

TO: Distribution
FROM: Joan Scott
DATE: July 13, 1976
SUBJECT: Approved MCR's

Attached are the Multics Change Requests which were approved from July 1, 1976 to July 15, 1976.

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

TITLE: Improve BOS RUNCOM SKIP Facility		STATUS	DATE
AUTHOR: Noel I. Morris		Written	6/8/76
-Coded in: <input type="checkbox"/> PL/I <input checked="" 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 checked="" 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)	Status	P 6/22/76 A 07/06/76
	Lib. Maint. Tools	Expires	01/06/77
	Sys. Anal. Tools	DOCUMENTATION CHANGES	
	Sys. Prog. Tools	Document	Specify One or More
	355	MPM (Vol, Sect.)	
	BOS	PLMS (AN #)	
	Salvager	MOSN (Sect.)	
	Ring Zero	MPAM (Sect.)	
	Ring One	MSAM (Sect.)	
	SysDaemon/Admin.		
Runtime			
User Cmnd/Subr.			
Objections/Comments:		Info Segs	
		Other (Name)	AM81 (MOH)
		None (Reason)	

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

Summary:

An extension to the RUNCOM SKIP mechanism is proposed. This extension will allow the selective execution of lines in the RUNCOM file.

Reasons:

The new input format for the SAVE and RESTOR commands makes it desirable to be able to use some lines of a SAVE or RESTOR runcom, delete other lines, and modify still others.

Detailed Proposal:

Modify the BOS RUNCOM SKIP Mechanism. This mechanism will continue to work as before. Each line of the runcom will be printed out. The operator's console keyboard will be unlocked to accept input. C will cause the line just typed to be executed and skip mode to be turned off. X will cause the line just typed to be executed and skip mode left in effect. End of message will cause the line just typed to be ignored and skip mode left in effect. X followed by a new command line will cause the line just typed to be replaced by the command line just input and skip mode left in effect.

... RUNCOM END card lease is found. The DATA is added to the BOS command directory, and the file is written on the disk.

RUNCOM PRINT name

The BCD file specified by name is printed on the operator's console.

RUNCOM SKIP name - args -

Successive lines of the runcom file are printed and the keyboard unlocked after each. If the ~~runcom line~~ is not to be executed, press EOM and the next line is printed. ~~Once the correct line is reached, type GO and execution begins with this line. The first line executed must be a command to BOS.~~

just printed

operator's console

If the line is to be executed, type "X". If the line is to be replaced

by another line, type "X" followed by the replacement line. If the line is to be executed followed by all subsequent lines in the runcom file, type "C".

RUNCOM RUN name

A runcom file may be executed by simply typing its name as a command to BOS. This forces the execution of "RUNCOM RUN name". (See BOS RUNCOM FILES above.)

RUNCOM PUNCH name

The BCD file name is punched on cards.

AM 81 Sect IV

TITLE: Update system exec_coms for MR4.0		STATUS	DATE
AUTHOR: T. Casey		Written	6/15/76
-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 checked="" type="checkbox"/> yes <input type="checkbox"/> no -Incompatible change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR _____		Status	P 01/22/76 A 07/06/76
		Expires	01/06/77
		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)	none

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

SUMMARY:

Merge changes made to the MIT versions of the system exec_coms since MR3.1 into the official versions; make sure the ones not used at MIT (e.g. asu.ec) still work.

Segments to be modified include:

- acct_start_up.ec (asu.ec)
- admin.ec
- biller.ec
- make_sys_seg.ec
- master.ec
- system_start_up.ec

DETAILED PROPOSAL:

admin.ec
 Change admin\$xxx to sc_command xxx.
 Change x io to start prta instead of prtb
 (most sites have a prta but not a prtb).

asu.ec

Fix up creation and setting of quotas on directories, to recognize that master directories may already have been created, and that project dirs and dirs off the root should have dir quotas as well as seg quotas.

Move generation of dds.absin to make_sys_seg.ec.

Fix up comments.

Fix minor bugs that cause disturbing error messages to be printed during running of asu.ec.

Bring up to date the setting-up of network data bases.

Change part 2 and the instructions for using it, to:

1. be run between "multics" and "go" instead of after "startup";
2. issue some "install" commands.

dds.absin

Change dvm -left to list_vols

make_sys_seg.ec

This ec generates admin.ec, dds.absin, and system_start_up.ec. The changes to these segments are listed separately.

Add dds.absin (it used to be generated by asu.ec).

Fix bug to avoid "substitute failed" messages.

master.ec

Add move_dir_quota to new_proj entry

Change device_meters to list_vols in disk_report entry.

system_start_up.ec

Remove set_max_length (no longer needed)

Change checks for existence of ".message" segments to only check for ones used by non-default acs.

Change admin\$xxx to sc_command xxx

Add set_sons_volume>pdd public

Delete set_timax 1

Add: delete_old_pdds

list_vols -tt

set_flagbox booting false

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>1988</u>
TITLE: fdump of FNP from Multics		STATUS DATE
AUTHOR: Robert S. Goren		Written 05/24/76
	MJG	Status 07/06/76
		Expires 12/24/76
Planned for System: MR 5.0		
Fixes Bug Number(s): not applicable		CATEGORY (check one)
Documented in MTB: not applicable		() Lib. Maint. Tools
Incompatible Change: no		(<input checked="" type="checkbox"/>) Sys. Anal. Tools
User/Operations-visible Interface Change: yes		() Sys. Prog. Tools
Coded in: (<input checked="" type="checkbox"/>) PL/I () ALM () other-see below		() 355
Performance: () better () same () worse		() BOS
		() Salvager
<u>DOCUMENTATION CHANGES (specify one or more)</u>		() Ring Zero
MPM (vol,sect) MPAM (sect)		() Ring One
MOSN (sect) MSAM (sect)		() SysDaemon/Admin
PLMs (AN#) AN51		() Runtime
Info Segs		() User Command/Subr
Other		
OBJECTIONS/COMMENTS:		

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

SUMMARY:

1. Install a subroutine, fdump_fnp_, in order to create an fdump of an FNP from Multics.
2. Install a command, online_dump_fnp, to output such a dump.

REASONS: Since it is now possible for an FNP to crash without crashing Multics, a mechanism is needed to obtain a dump of a crashed FNP other than by use of BOS commands. Since ERF numbers are tied to Multics crashes, a different convention has to be used for naming the segment containing such an FNP dump, and an online dumping command must be provided that understands the new naming convention.

IMPLICATIONS: There will now be two commands for "online-dumping" FNP dumps: the new online_dump_fnp, which handles dumps created by fdump_fnp_, and the already existing online_dump_355, which handles dumps created by the BOS command FD355.

DETAILED PROPOSAL: See attached draft Tools PLM documentation.

Name: fdump_fnp_

This subroutine dumps the contents of an FNP into a segment in >dumps, for later examination by means of online_dump_fnp.

Usage

```
declare fdump_fnp_entry (fixed bin, char (*), fixed bin (35));  
call fdump_fnp_ (fnp_no, entry_name, code);
```

fnp_no is the number of the FNP to be dumped (1, 2, 3 or 4).
 (Input)

entry_name is the entry name of the segment in >dumps created by
 fdump_fnp_. (Output)

code is a standard system status code. (Output)

- Notes:**
1. Use of this subroutine requires access to hphcs_.
 2. This subroutine cannot be used while the specified FNP is running, or while it is being loaded. If it is invoked under either of these circumstances, the error code error_table_\$io_still_assnd is returned.
 3. The entry name returned in entry_name is of the form fnp.TAG.DATE.TIME, where TAG is the FNP tag (a, b, c, or d), DATE is the current date in the form MMDDYY, and time is the current time in the form HHMM.

Names: online_dump_fnp, od_fnp

This command is used to output an ASCII dump of an FNP corresponding to a core dump in >dumps.

Usage

online_dump_fnp control_arqs

where control_arqs are selected from the following list:

- tag fnp_tag specifies the FNP tag component of the dump name (see Note below). fnp_tag must be one of the characters a, b, c, or d.
- date mmddyy,
-dt mmddyy specifies the date component of the dump name (see Note below). If this argument is not supplied, the current date is used.
- time hhmm,
-tm hhmm specifies the time component of the dump name (see Note below).
- pathname path,
-pn path specifies the pathname (relative or absolute) of the dump segment. This argument, if specified, overrides the -date, -time, and -tag arguments if any of them are supplied.
- dim dim_name specifies the Device Interface Module (DIM) to be used to output the dump. For reasons of compatibility, it must be an IOS-type DIM. This argument must be supplied.
- device device_name,
-dv device_name specifies the device to which the dump is to be output. This argument must be supplied.

Note: The name of an FNP dump segment is of the form fnp.TAG.DATE.TIME, where TAG, DATE, and TIME are as described above. In specifying the dump segment to online_dump_fnp (other than by using the -pathname control argument), the TAG and/or TIME component may be omitted if the remaining information is sufficient to uniquely identify the dump.

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>1991</u>								
TITLE: Allow io_call attach command to specify entry name	AUTHOR: Larry Johnson MJG	<table border="1"> <tr> <th>STATUS</th> <th>DATE</th> </tr> <tr> <td>Written</td> <td>06/24/76</td> </tr> <tr> <td>Status</td> <td><u>07/06/76</u></td> </tr> <tr> <td>Expires</td> <td>12/24/76</td> </tr> </table>	STATUS	DATE	Written	06/24/76	Status	<u>07/06/76</u>	Expires	12/24/76
STATUS	DATE									
Written	06/24/76									
Status	<u>07/06/76</u>									
Expires	12/24/76									
Planned for System: not applicable 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 type="checkbox"/> better <input checked="" type="checkbox"/> same <input type="checkbox"/> worse	<u>CATEGORY (check one)</u> <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 Command/Subr									
<u>DOCUMENTATION CHANGES (specify one or more)</u> MPM (vol,sect) cmd MPAM (sect) MOSN (sect) MSAM (sect) PLMs (AN#) Info Segs Other										
OBJECTIONS/COMMENTS:										

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

Summary

Allow the attach entry name to be specified in the io_call command. This would allow things like:

```
io_call attach sw foo$vfile_attach
```

This would perform a vfile_attachment, assuming some version of vfile_ were initiated with the reference name foo. Currently, io_call constructs the entry name by concatenating "attach" to the reference name, or entry name of the I/O module. Thus, initiating vfile_ with the reference name foo cannot work because io_call will look for foo\$fooattach.

Reasons

Currently, io_call will only make an attachment if the io_module can be initiated with the "correct" reference name. Since only one segment at a time can have this reference name, it is awkward to use two different versions of the same I/O module in a process. This is necessary in the case where you are testing a new version of an I/O module that is required normally by your process. An example would be an I/O module by which user_output is attached to user_i/o.

Implications

None. This sort of mechanism can already be used for testing commands.

Operation: attach

io_call attach switchname modulename -args-

where:

1. modulename is the name of the I/O module to be used in the attachment.
2. args may be one or more arguments, depending on what is permitted by the particular I/O module.

This command attaches the I/O switch using the designated I/O module. The attach description is the concatenation of modulename and args separated by blanks. The attach description must conform to the requirements of the I/O module.

If a control block for the I/O switch does not already exist, one is created.

If module name contains "<" or ">" characters, it is assumed to be a pathname, otherwise it is a reference name.

If the I/O modulename is specified by a pathname, it will be initiated with a reference name equal to the entry name. If the entry name or reference name does not contain a "\$", the attachment will be made by calling modulename#modulenameattach. If a "\$" is specified, the entry point specified will be called. See the MPM Reference Guide, "Entry Point Names".

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>1992</u>
TITLE: Change discard_ to reject the io_call order	AUTHOR: Larry Johnson	MJG
Planned for System: MR 5.0	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 () ALM () other-see below
Performance: () better (<input checked="" type="checkbox"/>) same () worse	STATUS DATE Written 06/24/76 Status A07/06/76 Expires 12/24/76	
DOCUMENTATION CHANGES (specify one or more)		CATEGORY (check one)
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
		() Ring Zero
		() Ring One
		() SysDaemon/Admin
		() Runtime
		(<input checked="" type="checkbox"/>) User Command/Subr
OBJECTIONS/COMMENTS:		

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

Summary

Currently, discard_ returns a code of 0 on all control calls. It should return error_table_\$no_operation on an io_call order.

Reasons

An I/O module that is attached to another I/O switch can implement the control operation by processing orders it recognizes and passing others it does not recognize on to the target switch. If the target switch is "discard_" and the order is "io_call" (from the io_call command) there is a problem. The io_call command was recently changed to issue an "io_call" order to perform control operations. If the I/O module does not recognize "io_call", the command then retries the original order. In the case described above, the "io_call" order appears to succeed (because discard_ returns 0), and the real order is never retried. The effect is that no orders are possible with the io_call command to such an I/O module if it is attached to a switch controlled by discard_.

Detailed Proposal

Have discard_ return error_table_\$no_operation on an "io_call" order. This will cause the io_call command to retry the original order.

Implications

None. Only the io_call command uses the io_call order.

Ver. 3	MULTICS CHANGE REQUEST	MTG	MCR 1993
TITLE: Permit hcs_\$status_long to return UID if process has access to target.			STATUS DATE
AUTHOR: Richard Bratt			Written 06/25/76
			Status A 07/06/76
			Expires 12/25/76
Planned for System: not applicable			CATEGORY (check one)
Fixes Bug Number(s): not applicable			() Lib. Maint. Tools
Documented in MTB: not applicable			() Sys. Anal. Tools
Incompatible Changes: no			() Sys. Prog. Tools
User/Operations-visible Interface Changes: yes			() 355
Coded in: (<input checked="" type="checkbox"/>) PL/I () ALM () other-see below			() BOS
Performance: () better (<input checked="" type="checkbox"/>) same () worse			() Salvager
DOCUMENTATION CHANGES (specify one or more)			(<input checked="" type="checkbox"/>) Ring Zero
MPM (vol,sect) subroutine MPAM (sect)			() Ring One
MOSN (sect) NSAM (sect)			() SysDaemon/Admin
PLMs (AN#)			() Runtime
Info Segs			() User Command/Subr
Other			
OBJECTIONS/COMMENTS:			

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

SUMMARY: If a process has non-null access to an object but no status permission to its parent, then hcs_\$status_long returns only some items. It is proposed that the UID of the object be added to the list of returned items.

REASONS: It seems perverse to allow a process to read or write a segment but not to admit the identity of the segment to the process.

IMPLICATIONS: None that I can think of.

Ver. 3	MULTICS CHANGE REQUEST	MJG	MCR 1994
741022			
TITLE: Provide a hardcore entrypoint which returns UID given segment number			STATUS DATE
AUTHOR: Richard Bratt			Written 06/25/76
			Status 07/06/76
			Expires 12/25/76
Planned for System: NR 5.0			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: yes			() 355
Coded In: (X) PL/I () ALM () other-see below			() BOS
Performance: () better (X) same () worse			() Salvager
DOCUMENTATION CHANGES (specify one or more)			(X) Ring Zero
MPM (vol,sect) subroutine MPAM (sect)			() Ring One
MOSN (sect) MSAM (sect)			() SysDaemon/Admin
PLMs (AN#)			() Runtime
Info Segs			() User Command/Subr
Other			
OBJECTIONS/COMMENTS:			

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

REASONS: In the current system a status long call is necessary to determine the identity of an object. This is rather heavy handed and needlessly expensive for many applications where a segment number which is bound to the object is available.

PROPOSAL: Provide a gate, hcs_\$get_uid_seg, which, given a segment number, returns the unique identifier of the designated object and a status code. This MCR does not wish to deal with the possibility of providing an hcs_\$get_uid_file entry point. It should be noted that this call will be extremely cheap since the KST, which is indexed by segment number, contains the UID of the designated object.

IMPLICATIONS: The efficiency of certain network and delayed deletion procedures can be increased.

Name: hcs_\$get_uid_seg

This entry returns the unique identifier of the object designated by a given segment number. This entry may return error_table_\$invalidsegno if the specified segment number is bound to no object and error_table_\$no_info if the process has not yet established its right to know that the designated object exists.

Usage: declare hcs_\$get_uid_seg entry (ptr), bit (36) aligned,
fixed bin (35));
call hcs_\$get_uid_seg (segptr, uid, code);

N.B. The reference to error_table_\$no_info in the above documentation will not appear in the MPM until directories may be initiated in the outer ring.

TITLE: Add -force option to delete and delete_dir		STATUS	DATE
AUTHOR: S. Herbst		Written	06/29/76
		Status	A 07/06/76
		Expires	12/29/76
-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 5.0		<input type="checkbox"/> Lib. Maint. Tools	
-Fixes Bug Number(s) _____		<input type="checkbox"/> Sys. Anal. Tools	
-Documented in MTB _____		<input type="checkbox"/> Sys. Prog. Tools	
-User/Operations-visible		355	
Interface change? <input checked="" type="checkbox"/> yes <input 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	
A separate MCR will be needed to remove delete_force		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:

Add -force control argument to delete and delete_dir. delete -force operates the same as the delete_force command. delete_dir -force deletes a directory whether or not its safety switch and copy switch is on, without issuing a query. Remove the delete_force command at some time in the future.

REASONS:

One fewer command in the system, where the force feature is properly an option to commands.

delete

delete

Name: delete, dl

The delete command causes the specified segments and/or multisegment files to be deleted. See also the descriptions of the delete_dir and delete_force commands (for deleting directories and deleting protected segments or multisegment files without being interrogated, respectively).

Usage

delete paths - *control-arg-*

where ^{1.} paths are the pathnames of the segments or multisegment files to be deleted.

2. *control-arg* can be -force to delete the specified entries whether or not they are protected, without issuing a query.

Notes

In order to delete a segment or multisegment file with the delete command, the entry must have its safety switch off and the user must have modify permission for the directory. If the safety switch is on, the user is interrogated as to whether he wishes to delete the entry. See also the description for the delete_force command to delete without interrogating the user.

If any one of the paths is a link, delete prints a message; it does not delete either the path in question or the link. (See the description of the unlink command.) If any one of the paths is a directory, delete prints a message; it does not delete the directory. (See the description of the delete_dir command.)

The star convention can be used.

delete_dir

delete_dir

Name: delete_dir, dd

The delete_dir command causes the specified directories (and any segments, links, and multisegment files they contain) to be deleted. All inferior directories and their contents are also deleted. See the descriptions of the delete and delete_force commands for an explanation of deleting segments and deleting protected segments, respectively.

Usage

delete_dir paths *-control_arg-*

where: ^{1.} paths are the pathnames of the directories to be deleted.

Notes
2. control_arg can be -force to delete the specified directories without issuing a query.

The user must have modify permission for both the directory and its superior directory. The star convention can be used. Before deleting each specified directory, delete_dir asks the user if he wants to delete that directory. It is deleted only if the user types "yes".

Warning: Protected segments in path_i or any of its subdirectories are deleted. Segments whose write bracket is less than the current ring are not deleted; consequently, the subtree being handled is not completely deleted if any such segments exist in the directory. For a discussion of ring brackets, see "Intraprocess Access Control (Rings)" in Section III of the MPM Subsystem Writers' Guide.

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>199 7</u>
TITLE: Install command run_new_fortran in experimental library.		STATUS: _____ DATE: 06/29/76
AUTHOR: Steve Webber		Written: _____ Status: <u>A 07/06/76</u> Expires: 12/29/76
Planned for System: MR 4.0		
Fixes Bug Number(s): not applicable		CATEGORY (check one)
Documented in MTB: not applicable		() 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: () better (X) same () worse		() BOS
		() Salvager
DOCUMENTATION CHANGES (specify one or more)		() Ring Zero
MPM (vol,sect) (EXL) MPAM (sect)		() Ring One
MOSN (sect) MSAM (sect)		() SysDaemon/Admin
PLMs (AN#)		() Runtime
Info Segs		(X) User Command/Subr
Other		
OBJECTIONS/COMMENTS:		

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

SUMMARY:

Install the commands run_new_fortran and locate_common in the experimental library.

REASONS:

Until the linker is changed to handle *system links in a reasonable way, an interim method of handling these linkage types must be provided. The new FAST Fortran compiler generates *system links for common so if a user wants to run programs compiled with this compiler that use common and yet does not want to run in FAST or DFAST he must use this command until the linker is changed.

IMPLICATIONS:

This program works as a run unit manager and hence may spoil normal Multics fortran users.

DETAILED PROPOSAL:

The program will set up a handler for linkage_error and will check to see if the link was a *system link. If so, a common block is allocated (if not already allocated) and initialized. Checks are made that all references to the same common block specify the same size unless the common block is "blank common". The command locate_common can be used to determine where the program has allocated the common blocks and how large they are. The common blocks are allocated in a temp seg which is released when the command returns. (Hence, locate_common only works when the run_new_fortran

stack frame is still active.

run_new_fortran

run_new_fortran

Name: run_new_fortran

EXPERIMENTAL LIBRARY --- TEMPORARY

The run_new_fortran command can be used to provide a limited "run unit" mechanism as well as the necessary system support for *system links until the system supports them in the intended manner.

A run unit is a limited execution environment that enables users to execute a FORTRAN or PL/I program repeatedly in a process having static storage and common blocks reinitialized with each invocation. PL/I external variables are also reinitialized.

The locate_common command (and active function) can be used in conjunction with run_new_fortran to display values allocated in the common storage created as a result of handling *system links.

Usage

run_new_fortran main-program-name -arguments-

where:

1. main-program-name is the reference name or relative pathname of the main program for the program run. (If main-program-name includes any ">"s or "<"s it is treated as a relative pathname.)
2. arguments are optional character string arguments to be passed to the main program.

Notes

The command works by getting temporary segments for common blocks (including unlabeled common) and setting up a handler for *system linkage faults which allocates common blocks in these temporary segments. The command releases the common blocks upon return (or as a result of a cleanup condition being signalled). Hence, if a user wants to look at the storage in a common block, he must somehow reach command level while the run_new_fortran command is still active. The locate_common command can be used in

run_new_fortran

run_new_fortran

this case.

The run_new_fortran command makes the main program specified in the command invocation unknown after the run. This is done even if the program was known prior to the run.

locate_common

locate_common

Name: locate_common

EXPERIMENTAL LIBRARY --- TEMPURARY

The locate_common command prints out the location and size of a common block allocated by the run_new_fortran command. It must be called when the run_new_fortran command is still active on the stack.

Usage

locate_common common_block_names -control_arg-

where:

1. common_block_names is a list of common block names to be located. The command prints out the size and location of each common block.
2. control_arg can be chosen from the following:
 - unlabeled to indicate that information about unlabeled common is wanted.
 - all (-a) to indicate that information about all common blocks is wanted.

Note

If no arguments are specified, information about unlabeled common is printed.

Example

```
locate_common a data -unlabeled
Common a at 347:40, 120 words.
Common data at 347:162, 4020 words.
Blank common at 350:0
```

```
r 630 0.193 0.014 2
```

locate_common

locate_common

Usage

[locate_common common-name]

where common-name is the name of a block of labeled common or is "-unlabeled". The character string returned consists of three octal numbers, the segment number, the offset within the segment, and the number of words of common. (These three values are just what the command dump_segment wants.)

Example

dump_segment [locate_common -unlabeled]

000220 000000001200 000000777774 666777666000 123123123123
000224 000000000000 000000000000

r 1332 0.564 12.773 4

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>1998</u>
TITLE: Reorganize disk-pack initializers, fix bugs. AUTHOR: Bernard Greenberg		STATUS DATE Written 06/29/76 Status <u>407/07/76</u> Expires 12/29/76
Planned for System: MR 5.0 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 type="checkbox"/> better <input checked="" type="checkbox"/> same <input type="checkbox"/> worse		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 checked="" type="checkbox"/> SysDaemon/Admin <input type="checkbox"/> Runtime <input type="checkbox"/> User Command/Subr
DOCUMENTATION CHANGES (specify one or more) MPM (vol,sect) MPAM (sect) MOSN (sect) MSAM (sect) PLMs (AN#) 61 Info Segs Other		
OBJECTIONS/COMMENTS:		

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

SUMMARY: Remove common code from `init_empty_root` and `init_disk_pack_` to separate program. Fix bugs in this path.

REASONS: Volume maps are not initialized properly, the last few bits of the last word of a volume map are not set right. Unused words are set to ones, not zeros. Identical code to set up volumes is shared by `init_empty_root`, which initializes RPV's on cold boots, and `init_disk_pack_`, which runs in ring 1 or ring 4 for all other disks. This duplicate code is a maintenance problem.

PROPOSAL: Create an active-all-rings subroutine which accepts an entry variable for a disk-writing routine, called by both of these programs. Reorganize volume map initializer to do it right.

IMPLICATIONS: `salvage_pv` also is malevolent about the last n bits of the volume map; change him to zero them.

(Sample)

MULTICS CHANGE REQUESTS BULLETIN

MCRB-XXX

To: MCR Distribution
From: Joan Scott
Date: XX/XX/XX
Subject: Approved MCR's from MCRXXX through MCRXXX.

Attached are the Multics Change Requests which were approved from XX/XX/XX through XX/XX/XX.

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

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR 1948
TITLE: Install format_blocks Utility Program		STATUS DATE
AUTHOR: David M. Jordan (EDS)		Written 06/02/76
Planned for System: not applicable		Status 06/15/76 A 7/13/76
Fixes Bug Number(s): not applicable		Expires 12/02/76 1/13/77
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) 2, III	MPAM (sect)	() Salvager
MOSN (sect)	MSAM (sect)	() Ring Zero
PLMs (AN#)		() Ring One
Info Segs		() SysDaemon/Admin
Other		() Runtime
OBJECTIONS/COMMENTS:		() User Command/Subr

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

SUMMARY:

Install a new utility program "format_blocks".

REASONS:

Users have occasional need to perform special purpose text formatting for such results as producing label output or other multi-column text. Currently, several single purpose programs exist to fulfill these needs.

IMPLICATIONS:

Single purpose programs may be eliminated. New applications making use of the generalized nature of format_blocks may be developed.

DETAILED PROPOSAL:

See attached user documentation.

format_blocks

format_blocks

Names: format_blocks, fb

EXPERIMENTAL LIBRARY --- TEMPORARY

The format_blocks command provides a generalized ability to rearrange a text file into fixed size blocks. Control options allow specification of such parameters as: number of columns, width of blocks, length of blocks, and spacing between blocks. In addition, some intra-block editing (indentation, centering, use or non-use of null lines) is allowed.

format_blocks does not provide the complex formatting capabilities allowed by programs such as runoff. In particular: there are no per-block controls, all controls apply to all blocks; there is no capability for changing block specifications during processing as all options are provided by command arguments, and there are no provisions for such items as headers and footers. Many of these functions may be accomplished, however, by using runoff to format input for format_blocks or by using runoff to reprocess format_blocks output.

Usage

format_blocks input_file output_file -control_args-

where:

1. input_file

is the pathname of the file containing text to be formatted.

2. output_file

is the pathname of the file into which formatted text is to be placed.

3. control_args

may be taken from the following lists:

Internal Block Format Control Arguments

-width *n*

Specifies block width in print positions. Minimum width is 1 position, maximum is 200 positions. Default width is 40 print positions.

- length n , -ln n**
Specifies block length in lines. Minimum length is 1 line, maximum is 200 lines. Default length is 8 lines.
- indent n , -ind n**
Specifies that each line is to be indented n print positions within its block. Minimum indentation is 0 positions, maximum is $\min(200, \text{width}-1)$. Default indentation is 0 print positions.
- center, -ce**
Specifies that lines are to be horizontally centered with the blocks. Note that the default is to indent 0 positions rather than to center.

Input Processing Control Arguments

- delimiter XX, -dm XX**
Specifies that the string XX is to be interpreted as a break between blocks of input. Any input line beginning with the string XX will be treated as a break. Any other characters on the break line will be ignored. Minimum length of the break string is 1 character, maximum length is 200 characters. The default break string is "!!!".
- no_edit, -ned**
Specifies that null input lines are to be included in the output. The default is to remove null lines from the input before formatting the output. See also "Notes" below, as this control argument also affects the vertical positioning of lines within a block.
- no_discard, -ndcd**
Specifies that an implicit break is to occur if a block has been filled but no break has been found. A new block will be begun with the implicit break and will be continued until the next break or implicit break. The default is to discard any input lines encountered after a block has been filled but before an explicit break has been found.

Block Layout Control Arguments

- columns n , -cols n**
Specifies that blocks are to be formatted in n columns. The minimum number of columns is 1, the

format_blocks

format_blocks

maximum is 200. The default number of columns is 3.

-gutter *n*
Specifies that *n* character positions are to be inserted between columns of output. The minimum gutter is 0 character positions, the maximum is 200. The default gutter is 1 character position.

-lines_between *n*, -lbt *n*
Specifies that *n* lines are to be inserted between blocks of output. The minimum number of lines is 0, the maximum is 200. The default number of lines is 1.

Output Processing Control Arguments

-extend, -ext
Specifies that output should be appended to a previously existing output file. The default is to overwrite any previously existing file.

-header, -he
Specifies that a block outline (formatted by taking **-width**, **-length**, **-columns**, **-gutter**, and **-lines_between** into account) should be outputted several times before actual processing begins. **format_blocks** will output this block outline until a block is completed after 196 lines of output, thus resulting in about three pages of outlines. This feature is useful if the output is to be printed on special forms (such as label stock) as it allows the operator to properly align the forms before actual data is printed. The default is not to print a header.

Error and Status Printing Control Arguments

-brief, -bf
Specifies that non-fatal errors (see "Notes" below for descriptions of non-fatal errors) are not to be reported as they are encountered. The default is to report each error as it is detected.

-totals, -tt
Specifies that a summary of processing (see "Notes" below) is to be printed after processing has completed. The default is to not print a summary.

Notes

1. Vertical Spacing of Lines within Blocks

Vertical spacing of lines within blocks is dependent on the `-no_edit` control argument. If `-no_edit` has not been specified, lines will be centered vertically in each block. If `-no_edit` has been specified, lines will be placed in a block starting at the top of the block and unused lines will be blanked out.

2. Error and Status Reporting

Four types of non-fatal error are detected in the input:

- a) Input line exceeds maximum buffer size.
format_blocks allows buffer space enough for two characters in each output print position. This error occurs if an input line exceeds the maximum buffer size.
- b) Input line exceeds block width.
This error occurs if an input line contains more than $(width - indent)$ print positions.
- c) Input line attempts to backspace past print position 1.
- d) Input block exceeds output block length.
This error occurs if `-no_discard` has not been specified and more than "length" lines are encountered before the next break line.

Each of the above errors is non-fatal, but results in some loss of data between input and output.

In `-brief` mode, the above errors are not reported as they are encountered. Otherwise, the above errors are reported and the line number of the offending input line is also reported.

If `-totals` has been specified, a report will be printed after processing which will include the number of input lines processed, the number of output blocks formatted, the total number of non-fatal errors encountered, and a break down of errors by the above types.

3. Conflicting Control Arguments

If conflicting control arguments (such as `-center` and `-indent`) or multiple instances of a single control argument are found in the command invocation, format_blocks will

format_blocks

format_blocks

report an error and abort processing.

4. Handling of Special Characters in the Input File

format_blocks considers only three characters (new_line, tab and backspace) as special. All other characters appearing in the input are assumed to take exactly one print position and to not affect line spacing. Thus, it is the user's responsibility to assure that non-printing characters and characters which affect printing (new_page, for instance) are not present in the input.

Examples

Assuming that a user had a file "labels.input" with the following contents:

Ms. Kathleen Johnson
234 Anystreet, Apartment 34
Boston, MA
02115

!!!
Mr. Howard Johnson
23 Flavor St.
Phoenix, AZ
!!!

Mr. John Smith
Plymouth, MA
!!!

The Editor
Time Magazine
Room 5078
51 W. 34th St.
New York, NY

And that the user wanted to format the data appropriately for printing on label stock with the following characteristics: 2 labels across, 40 characters/label, 1 character between labels, 8 lines/label, and 1 line between labels; and that a header was desired in order to allow the operator to properly align the label stock; the following command could be used to produce the desired output:

```
format_blocks labels.input labels.output -columns 2 -header
```

After execution of the above command, "labels.output" would be as appears on the next page.

format_blocks

format_blocks

```
*****
*
*
*
*
*
*
*****
*
*
*
*
*
*
*****
```

Ms. Kathleen Johnson
234 Anystreet, Apartment 34
Boston, MA
02115

Mr. Howard Johnson
23 Flavor St.
Phoenix, AZ

Mr. John Smith
Plymouth, MA

The Editor
Time Magazine
Room 5078
51 W. 34th St.
New York, NY

As another example, assume that a user wanted to format a text file such that it would print two columns of text on a page. Assume further that the columns would be 38 characters wide, with 4 characters between columns and that each column would be 66 lines long. Also, assume that the text is to be generated from a runoff input file. To obtain the desired result, the user could:

- 1) edit the original runoff file to add a ".ll 38" control line at the beginning; execute the following runoff commands:

```
runoff text -segment -no_pagination -indent 0
```

followed by the following format_blocks command:

```
format_blocks text.runout text.output -no_edit -no_discard -width  
38 -length 62  
-columns 2 -gutter 4 -lines_between 9
```

format_blocks

format_blocks

The above sequence would result in output like that on the next page.

1. Introduction

This document is a semi-formal definition of the language supported by the Multics PL/I compiler. The document is intended to be used as a reference manual by programmers who need exact answers to detailed questions concerning the syntax and semantics of Multics PL/I. In keeping with that purpose, the document defines the language in an analytic rather than a synthetic manner; i.e. it explains the meaning of programs, but does not describe how to construct programs.

Additional information useful to the Multics PL/I programmer is found in the following documents:

The "Multics PL/I User's Guide" provides an introduction to Multics PL/I, gives guidance on how to write a Multics PL/I program, and explains the relationship between Multics PL/I and the run-time environment supplied by the Multics system.

The "Multics Programmers' Manual" describes the Multics system and includes brief descriptions of each command and subroutine available on the system.

The "Subsystem Writer's Guide" contains detailed descriptions of the representation of PL/I data, the exact layout of a PL/I activation record (stack frame), the internal format of a PL/I area, the calling sequence generated for a PL/I call, etc. Most users should never require this information.

1.1 The Language

1.2 The Method of Definition

The language is defined using a meta-language to define the syntax in prose to describe the semantics. Although this is a semi-definition, both the syntactical and semantic descriptions are as precise and complete as possible.

Example:

```
<based attribute> ::=  
  based[( <locator reference
```

When the prose refers to a <attribute> or a <locator reference>, these terms appear exactly as they do in the syntax rule. When a keyword appears in prose, it is enclosed in quotes to distinguish it from text; for example, "based" and "float".

Terms defined in prose are underlined when defined and not underlined thereafter. Examples are provided to aid understanding but are not intended to be comprehensive or definitive. All examples are clearly set off from the rest of the text as shown by the example on this page. Within examples, where empty space might be misleading, a blank space denotes a blank.

1.2.1 The Meta-Language

The syntax of the PL/I language is defined by a set of syntax rules expressed in a formal notation derived from Backus-Naur Form. Each rule describes a character-string pattern of characters that constitute a syntactic construct of the language. The complete set of rules describes all syntactically correct PL/I programs.

Example:

format_blocks

The Multics PL/I language is a dialect of the proposed ANSI/ECHA Standard PL/I. Because the PL/I standard was not complete when this document was published, the exact differences between the two languages could not be defined. However, the languages are so similar that nearly all Multics PL/I programs are valid programs in standard PL/I.

format_blocks

```
<skip option> ::=  
    skip[(<expression>)]
```

In this example, *<skip option>* notation variable that represent character-string described by syntax expression on the right definition symbol " ::= ". "skip notation constant that represent actual occurrence of character-string "skip". *<expression>*

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR 2001
TITLE: Install usase_and_revenue command		STATUS: DATE
AUTHOR: Jim Homan	TW	Written: 06/28/76 Status: A07/13/76 Expires: 12/28/76
Planned for System: not applicable		
Fixes Bus Number(s): not applicable		
Documented in MTB: 279		
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 type="checkbox"/> better <input checked="" type="checkbox"/> same <input type="checkbox"/> worse		
DOCUMENTATION CHANGES (specify one or more)		
MPM (vol,sect) MPAM (sect)		
MOSN (sect) MSAM (sect) 3		
PLMs (AN#)		
Info Sess		
Other		
OBJECTIONS/COMMENTS:		

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

SUMMARY

Install the usase_and_revenue command as described in MTB-279, with the exception that the "Space on logical volumes" section has been deleted, primarily because it duplicates the function of the list_vols command.

REASON

Provide a tool for reporting on system usage and revenue.

IMPLICATIONS

None.

```
-----  
| usage_and_revenue |  
|-----|
```

name: usage_and_revenue

The usage_and_revenue command prints out a report of system usage and revenue broken down by groups of users.

Usage

usage_and_revenue control data -data2-

- 1) control is the pathname of an ASCII file which defines the groups of users for the report.
- 2) data is the pathname of a copy of the system use_totals month-to-date statistical database.
- 3) data2 is the pathname of an earlier copy of the system use_totals month-to-date statistical database.

Notes

If both the data and data2 arguments are given, a daily report will be produced showing the incremental system usage between the time data2 was created and the time data was created. If only the data argument is given then a monthly report will be produced showing the month-to-date usage at the time data was created.

The control file may specify up to 9 groups of users to be shown in the report. Each group consists of 1 or more of the usage bins in the use_totals database. The groups will appear in the report in the same order in which they appear in the control file. Each line of the control file specifies one of the groups (except that lines beginning with a "*" are ignored). The format of the lines is:

Group title:users1,users2,...,usersn

Where

Group title is the title which will be used to identify the group in the report. The maximum length is 24.

usersn are either

- 1) the titles of the usage bins in use_totals which are to be included in this group. These are the titles specified in the control file for reset_use_totals. (See the MPM description of reset_use_totals.)

or

2) if usersn is enclosed in quotes, then it specifies another group defined in the control file and all the bins which go into that group will also go into this group.

Example:

If use_totals has bins labelled "Staff", "Users" and "Other", the control file

```
Staff Use:Staff
Non-Staff use:Users,Other
Total:"Staff Use","Non-Staff Use"
```

would result in a report with 3 groups, one containing one bin's usage (Staff), one containing 2 bins' usages (Users+Other), and one containing 3 bins' usages (Staff+Users+Other).

A facility is provided to add notes or messages to the report. If a segment named usage_and_revenue_footnote exists in the caller's working directory, its contents will be printed at the end of the report.

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>2002</u>
TITLE: Reduce size of multiseqment area seqs to 64K		STATUS DATE
AUTHOR: Garv C. Dixon	JWG	Written 06/30/76
		Status A 07/13/76
		Expires 12/30/76
Planned for System: MR 5.0		CATEGORY (check one)
Fixes Bug Number(s): unreported		<input checked="" type="checkbox"/> Lib. Maint. Tools
Documented in MTB: not applicable		<input 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 type="checkbox"/> other-see below		<input type="checkbox"/> BOS
Performance: <input type="checkbox"/> better <input checked="" 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
MCSN (sect) MSAM (sect)		<input type="checkbox"/> SysDaemon/Admin
PLMs (AN#) AN51, Tools		<input type="checkbox"/> Runtime
Info Seqs		<input type="checkbox"/> User Command/Subr
Other		
OBJECTIONS/COMMENTS:		

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

SUMMARY:

Initialize the area segments in a multiseqment area (MSA) so that they fit in, and are limited to, a 64K segment.

REASON:

MSAs are MSFs whose components are area segments. As with MSFs, an MSA begins as a single segment area (SSA) which grows into an MSA when its area overflows. This growing requires that an MSF directory be created, and that the SSA be copied as the first component in the MSF directory. This growing process causes problems when the SSA is in the process directory, where quota limitations prevent making a copy of a 255K segment. Since it is desirable for the library tools to create MSAs in the process directory, the maximum SSA size must be limited. It seems reasonable to use 64K as the limit, since a larger value would cause more 256K AST entries to be used.

IMPLICATIONS:

It will be possible to grow an SSA created in the process directory; therefore, new library tools will work properly. System performance may improve slightly as contention for 256K ASTEs is reduced. Users of MSAs larger than 64K may notice a slight degradation of performance due to the extra signalling of area condition, and conversion of SSA to MSA.

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>2003</u>
TITLE: Install subroutine interface to TCT and TCTR instructions		STATUS DATE Written 06/30/76
AUTHOR: Gary C. Dixon	JWG	Status A 07/13/76 Expires 12/30/76
Planned for System: MR 5.0		
Fixes Bug Number(s): not applicable		CATEGORY (check one)
Documented in MTB: not applicable		() Lib. Maint. Tools
Incompatible Change: no		() Sys. Anal. Tools
User/Operations-visible Interface Change: yes		(<input checked="" type="checkbox"/>) Sys. Prog. Tools
Coded in: () PL/I (<input checked="" type="checkbox"/>) ALM () other-see below		() 355
Performance: () better (<input checked="" type="checkbox"/>) same () worse		() BOS
		() Salvager
DOCUMENTATION CHANGES (specify one or more)		() Ring Zero
MPM (vol,sect) MPAM (sect)		() Ring One
MOSN (sect) MSAM (sect)		() SysDaemon/Admin
PLMs (AN#) AN51, tools		() Runtime
Info Segs		() User Command/Subr
Other		
OBJECTIONS/COMMENTS:		
Improve documentation to really explain how to make up the table.		

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

SUMMARY: Install the tct_ subroutine.

REASONS:

The PL/I search and verify built-in functions have several deficiencies which make certain operations impossible to perform. The BIFs use a TCT translate table which contains only 128 characters, corresponding to the ASCII character set. Therefore, the results of searching or verifying a character string containing nonascii characters is nondeterminate. This deficiency makes it impossible to verify that a character string contains only ASCII characters.

The search and verify BIFs simulate the operation of of the TCT and TCTR instructions when the list of search or verify characters is not constant. For certain applications in which a search or verify is repeatedly performed with the same, nonconstant search characters, the increased cost of this simulation may be unacceptable.

Finally, the search and verify BIFs simulate the TCTR instruction (reverse searches) by operating on a reversed copy of the searched string which is created in the user's stack. If the searched string is large, this strategy can result in excessing paging, stack overflows, and it make be totally impossible (eg, searching an entire 255K segment).

The tct_ subroutine solves all of these problems by using a full, 512 character translate table with the TCT or TCTR Instructions.

IMPLICATIONS:

Programs which must perform the functions described above, such as the library maintenance tools, will be able to perform them efficiently.

tct_

tct_

Name: tct_

This subroutine uses the EIS translate character and test (TCT) instruction to perform the function of the PL/I search and verify built-in functions with a user-provided translation table. Unlike the PL/I built-ins, it allows (requires) translation tables with entries for characters outside the ASCII character set. Thus, it can be used to search for non-ASCII characters in a string.

In addition, the PL/I built-ins simulate the search and verify functions when the list of search characters is not a constant, rather than building a translate table at run time. The simulation is performed by indexing into the list of search (or verify) characters for each character in the string until (or while) a match is found. It is performed on a reversed copy of the string created in the stack when the reverse built-in is used in combination with search or verify. For some applications in which a repeated search or verify is performed with the same, nonconstant list of search characters, it may be more efficient for the program to construct its own translate table and call tct_. Even when the list of search characters is changing, if the searched string is long and must be searched many times, or if it is long and a reverse search or verify is required, it may be more efficient to use the entry points below which perform a search or verify operation by: constructing a run time translate table; using the TCT instruction.

Calls to tct_ normally search a string from left to right, returning a character index from the beginning of the string to identify the character found by the search. In order to be able to search from right to left, each type of tct_ call has a corresponding entry point (with "reverse" in its entry point name) that performs the search function from right to left, returning a character index from the end of the string.

Entries: tct_, tct_\$reverse

These two entry point take a searched string and a translation table as arguments, and return the index of the character in the string found by the search. See "Translate Table Format" below for a description of how to construct a translation table.

tct_

tct_

Usage

```
declare tct_ entry (char(*), char(512))
         returns (fixed bin(21));
```

```
index = tct_ (string, table);
```

where:

1. string is the character string to be searched. (Input)
2. table is the translation table. (Input)
3. index is the index of the character in the string which was found by the search. (Output) A value of 0 is returned if the search fails.

Entries: tct_\$translate, tct_\$reverse_translate

These entry points operate like tct_ and tct_\$reverse, but also return the value in the translate table which corresponds to the character which was found in the searched string.

Usage

```
declare tct_$translate entry (char(*), char(512),
                              fixed bin(21)) returns (char(1));
```

```
table_value = tct_$translate (string, table, index);
```

where:

1. string is as above.
2. table is as above.
3. index is as above.
4. table_value is the character in the translation table entry which corresponds to the character found in the searched string. (Output)

tct_

tct_

Entries: tct_\$search, tct_\$reverse_search

These two entry points perform the function of the PL/I search and reverse built-in functions shown below.

```
index = search (string, search_list);
```

```
index = search (reverse(string), search_list);
```

They perform this function by constructing a translate table from the search_list, and using the TCT or TCTR instruction to search the string with this table.

Usage

```
declare tct_$search entry (char(*), char(*))  
        returns (fixed bin(21));
```

```
index = tct_$search (string, search_list);
```

where:

1. string is as above.
2. search_list is the list of characters being searched for.
(Input)
3. index is as above.

Entries: tct_\$verify, tct_\$reverse_verify

These two entry points perform the function of the PL/I verify and reverse built-in functions shown below.

```
index = verify (string, verify_list);
```

```
index = verify (reverse(string), verify_list);
```

They perform this function by constructing a translate table from the verify_list, and using the TCT or TCTR instruction to search the string with this table.

tct_

tct_

Usage

```
declare tct_$verify entry (char(*), char(*))  
        returns (fixed bin(21));
```

```
index = tct_$verify (string, verify_list);
```

where:

1. string is as above.
2. verify_list is the list of characters to be verified as the contents of the string. (Input)
3. index is as above.

Translate Table Format

A translate table is a fixed length character string of 512 characters, one character for each of the possible 9-bit byte values. The table entry corresponding to a particular byte value is located by treating the 9-bit byte as a fixed binary(9) number, then by using this number as an offset (as opposed to a character index) into the table.

TITLE: Fix IOI Race Condition		STATUS	DATE
AUTHOR: Noel I. Morris		Written	6/30/76
-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	Status	A 07/13/76
	<input type="checkbox"/> Sys. Anal. Tools	Expires	01/13/77
	<input type="checkbox"/> Sys. Prog. Tools	DOCUMENTATION CHANGES	
<input type="checkbox"/> 355	Document	Specify One or More	
<input type="checkbox"/> BOS	MPM (Vol, Sect.)		
<input checked="" 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	Info Segs		
<input type="checkbox"/> User Cmnd/Subr.	Other (Name)		
	None (Reason)		
Objections/Comments:			

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

Summary:

A race condition in the I/O interfacer sometimes allows a user to issue a connect for a device after he has been given status for a previous operation but before the I/O Interfacer has cleared a device active bit. This condition occurs chiefly on multiple CPU systems and is aggravated by the logging of I/O errors after status has been delivered to a user and a wakeup sent.

Proposal:

On error conditions, log the status before delivering it to the user and sending a wakeup. In the connect code, loop for a short while (less than 1 millisecond) if the device active bit is still on. If the bit remains on, return an error code.

Implications:

backup will no longer get spurious errors causing it to die.

TITLE: Change BOS to print its version ID		STATUS	DATE
AUTHOR: Noel I. Morris JWG		Written	6/30/76
-Coded in: <input type="checkbox"/> PL/I <input checked="" type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL -Planned for System MR 4.1 -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	A 07/13/76
	<input type="checkbox"/> Lib. Maint. Tools	Expires	01/13/77
	<input type="checkbox"/> Sys. Anal. Tools	DOCUMENTATION CHANGES	
<input type="checkbox"/> Sys. Prog. Tools	355	Document	Specify One or More
<input checked="" type="checkbox"/> BOS	Salvager	MPM (Vol, Sect.)	
<input type="checkbox"/> Ring Zero	Ring One	PLMS (AN #)	
<input type="checkbox"/> SysDaemon/Admin.	Runtime	MOSN (Sect.)	
<input type="checkbox"/> User Cmmd/Subr.		MFAM (Sect.)	
		MSAM (Sect.)	
Objections/Comments:		Info Segs	
		Other (Name)	
		None (Reason)	

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

Reason:

There is currently no way to determine what version of BOS is in use.

Proposal:

Add a segdef to setup to allow the insertion of a system ID when generating the BOS system tape.' Change the message

BOS AT 11:38 WED

to

BOS 2.03 AT 11:38 WED

Implications:

New message from BOS which is slightly longer than the old message.

TITLE: Make BOS Bootloading Mechanism Reliable		STATUS	DATE
AUTHOR: Noel I. Morris <i>JWG</i>		Written	6/30/76
-Coded in: <input type="checkbox"/> PL/I <input checked="" type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Status	A 07/13/76
		Expires	01/13/77
-Planned for System MR _____		DOCUMENTATION CHANGES	
-Fixes Bug Number(s) _____		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 checked="" type="checkbox"/> no		MPAM (Sect.)	
-Performance: <input checked="" type="checkbox"/> Better <input type="checkbox"/> Same		MSAM (Sect.)	
<input type="checkbox"/> Worse		Info Segs	
-Replaces MCR _____		Other (Name)	
Objections/Comments:		None (Reason)	

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

Summary:

When using the self-loading BOS system tape, it is sometimes necessary to attempt to boot it several times in order to successfully bring up BOS. It has been discovered that the reason for bootload failures is the occurrence of spurious lockup faults when the tape is booted.

Proposal:

Modify the BOS module FWLOAD to catch lockup faults when the tape is bootloaded.

Implications:

The self-loading BOS tape will boot successfully every time.

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>2008</u>								
TITLE: Add if/then/else and case feature to ioa_	AUTHOR: Larry Johnson	RAE								
Planned for System: MR 5.0	Fixes Bug Number(s): not applicable	Documented in MTB: not applicable								
Incompatible Change: no	User/Operations-visible Interface Change: no	Coded in: ()PL/I (<input checked="" type="checkbox"/>)ALM ()other-see below								
Performance: ()better (<input checked="" type="checkbox"/>)same ()worse	<table border="1"> <tr> <td>STATUS</td> <td>DATE</td> </tr> <tr> <td>Written</td> <td>07/05/76</td> </tr> <tr> <td>Status</td> <td><u>07/13/76</u></td> </tr> <tr> <td>Expires</td> <td>01/05/77</td> </tr> </table>		STATUS	DATE	Written	07/05/76	Status	<u>07/13/76</u>	Expires	01/05/77
STATUS	DATE									
Written	07/05/76									
Status	<u>07/13/76</u>									
Expires	01/05/77									
DOCUMENTATION CHANGES (specify one or more)		CATEGORY (check one)								
MPM (vol,sect) AG93	MPAM (sect)	() Lib. Maint. Tools								
MOSN (sect)	MSAM (sect)	() Sys. Anal. Tools								
PLMs (AN#)		() Sys. Prog. Tools								
Info Segs		() 355								
Other		() BOS								
		() Salvager								
		() Ring Zero								
		() Ring One								
		() SysDaemon/Admin								
		() Runtime								
		(<input checked="" type="checkbox"/>) User Command/Subr								
OBJECTIONS/COMMENTS:										

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

Summary

Add features to ioa_ (formline_) to implement if/then/else and case constructs.

Reasons

Provides useful features to ioa_ (see Examples.)

Detailed Proposal

Draft MPM documentation on next page.

Draft ioa Documentation

`^[` starts an if/then/else or case selection group. A `^[` takes a fixed binary or a bit string argument, and must have a matching `]^` to limit its scope. The text between the `^[` and the `]^` may be divided into clauses delimited by `^;`. If `^[` is given a fixed binary argument of `n`, the `n`th clause between the `^[` and the `]^` is expanded; all other clauses are ignored. If there is no `n`th clause (`n` too large, or `<1`), all the text between the `^[` and the `]^` is ignored. If the argument to `^[` is a non-zero bit string, the first clause is expanded (equivalent to a fixed bin argument of 1, or "then"). If the argument to `^[` is an all-zero bit string, the second clause is expanded (the "else" case). `^[` controls may be nested up to 4 deep. Null clauses are permitted.

`]^` limits the scope of a `^[`. See `^[` above.

`^;` used a clause delimiter between `^[` and `]^`. See `^[` above. `^n;` is equivalent to `n` repetitions of `^;`.

Examples

```
Source:  sw="0"b;
         call ioa_ ("a=~d ^[b=~d^;^s^] c=~d",5,sw,7,9);
```

```
Result:  a=5 c=9
```

```
Source:  sw="1"b; /* Using same ioa_ call */
```

```
Result:  a=5 b=7 c=9
```

```
Source:  dir=">"; ename="foo";
         call ioa_ ("Error in segment ^a^[>^]~a",
                  dir, (dir ^= ">"), ename);
```

```
Result:  Error in segment >foo
```

```
Source:  dir=">foo"; ename="bar";          /* Using same ioa_ call */
```

```
Result:  Error in segment >foo>bar
```

```
Source:  option=2; /* Assume following call is on one line */
         call ioa_ ("Insurance option selected:
                  ^[no fault^;bodily injury^;property damage^]",
                  option);
```

```
Result:  Insurance option selected: bodily injury
```

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>2009</u>
TITLE: Upgraded version of expand_path_.		STATUS : DATE
AUTHOR: Bernard Greenberg		Written : 06/29/76
Planned for System: MR 5.0		Status : A07/13/76
		Expires : 12/29/76
Fixes Bug Number(s): not applicable		CATEGORY (check one)
Documented in MTB: not applicable		() 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: () better (X) same () worse		() BOS
DOCUMENTATION CHANGES (specify one or more)		() Salvager
MPM (vol,sect) subrs	MPAM (sect)	() Ring Zero
MOSN (sect)	MSAM (sect)	() Ring One
PLMs (AN#)		() SysDaemon/Admin
Info Segs		() Runtime
Other		(X) User Command/Subr
OBJECTIONS/COMMENTS: Warning to users is needed that this change is coming in case they replace expand_path_		

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

SUMMARY: Replace expand_path_ with a functionally equivalent module with a reasonable calling sequence.

REASONS: expand_path_ is one of the most popular interfaces in the system, and one of the first interfaces seen by one writing programs for Multics. The calling sequence of this module involves passing pointers to aligned strings: this is rooted in the inadequacies of EPL. Furthermore, the necessity of having aligned strings prohibits them from being passed legally by reference to the storage system interfaces.

Furthermore, no checks can be made that pointers so given point to strings of any given length; stack clobbering can occur by passing pointers to strings shorter than 168 and 32. Many "other" languages on Multics (e.g., LISP) cannot support passing character strings via pointers: the Multics standard is to pass character strings as such. The current expand_path_ interface is an abomination and should be relegated to obsolete status.

IMPLICATIONS: Clearer user interface. SWG documentation of old interface. expand_path_ supports a special function when the entry-string pointer is passed as null: a new interface (see below) must be provided for this function.

DETAILED PROPOSAL: Implement expand_pathname_ and absolute_pathname_ as below. Thanks to Bob Frankston for inventing and implementing this interface.

Name: expand_pathname_

This subroutine is used to convert a relative or absolute pathname into a directory name and entry name.

Usage: dcl expand_pathname_ entry (char(*), char(*), char(*), fixed bin (35));

call expand_pathname_ (relpath, dir, ent, code);

where

relpath is a relative or absolute pathname to be expanded (Input).

dir is the directory portion of the relative or absolute pathname given (Output).

ent is the entry name derived from the relative or absolute pathname given (Output).

code is a status code telling whether or not the pathname was successfully expanded. If 0, the pathname was successfully expanded. Other possible values for code are error_table_\$lesserr, error_table_\$badpath, error_table_\$pathlong, error_table_\$entlong as befits the semantics of these errors (Output).

Name: absolute_pathname_

This entry point is used to convert a relative or absolute pathname into an absolute pathname.

Usage: declare absolute_pathname_ entry (char(*), char(*), fixed bin (35));

call absolute_pathname_ (relpath, abspath, code);

where

relpath is the relative or absolute pathname to be expanded (Input).

abspath is the absolute pathname derived from relpath (Output).

code is a status code telling of the relative success of the expansion. If 0, the expansion was successful. If not, it is one of error_table_\$lesserr or error_table_\$pathlong as befits the semantics of these errors. (Output).

Ver. 3 741022	MULTICS CHANGE REQUEST	MCR <u>2010</u>
TITLE: Fix bug in load_vol_map		STATUS: DATE
AUTHOR: Bernard Greenberg	TVV	Written: 07/05/76
		Status: <u>A 07/13/76</u>
		Expires: 01/05/77
Planned for System: MR 5.0		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: (X) PL/I () ALM () other-see below		() BOS
Performance: () better (X) same () worse		() Salvager
DOCUMENTATION CHANGES (specify one or more)		(X) Ring Zero
MPM (vol,sect) MPAM (sect)		() Ring One
MOSN (sect) MSAM (sect)		() SysDaemon/Admin
PLMs (AN#) 61		() Runtime
Info Seqs		() User Command/Subr
Other		
OBJECTIONS/COMMENTS:		

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

SUMMARY: pvte.vtoc_size is set incorrectly by load_vol_map. pvte.n_vtoce, a potentially more useful field, is not set at all. Set these quantities correctly.

REASONS: Total vtoces are misaccounted by the outer ring because of this, and phcs_sget_vtoce fails for the last record of VTOC.

IMPLICATIONS: n/a.

Ver. 3
741022

MULTICS CHANGE REQUEST

MCR 2011

TITLE: Lay traps for trailer problems.

STATUS: _____ DATE: _____
Written: 07/05/76
Status: A 07/13/76
Expires: 01/05/77

AUTHOR: Bernard Greenberg

TVV

Planned for System: MR 5.0
Fixes Bug Number(s): not applicable
Documented in MTR: not applicable
Incompatible Change: no
User/Operations-visible Interface Change: no
Coded in: ()PL/I ()ALM ()other-see below
Performance: ()better ()same ()worse

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

DOCUMENTATION CHANGES (specify one or more)

MPM (vol,sect) MPAM (sect)
MOSN (sect) MSAM (sect)
PLMs (AN#)
Info Secs
Other n/a

OBJECTIONS/COMMENTS:

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

SUMMARY: Several crashes have occurred at MIT indicating possible mismanagement of the system trailer segment. The phenomenon is not understood at all. Set two debugging aids to help locate this problem should it manifest itself again.

REASONS: 1) At least twice, the end of the trailer free list has been hit, but in fact, a large number of free trailers existed. 2) At least twice, an SDW has been found in the Descriptor Segment of a process being destroyed, with no trailer to account for it. Although not much down time has been accrued due to these problems, they seem to be symptomatic of some malaise in segment control.

DETAILED PROPOSAL: 1) Have seq_fault crash when it attempts to use the last trailer on the free list. Since it would have crashed one trailer later anyway, this is not a major limitation. 2) Leave the "dir_seq" abs_seq covering the KST of a defunct process, so that when a missing-trailer crash occurs, we can at least hope to ascertain some facts about the identity of the guilty segment.

TITLE: Install new fortran compiler and command program in >ldd>exl>o		STATUS	DATE
AUTHOR: D. Levin		Written	07/06/76
SHW		Status	P 7/13
-Coded in: <input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Expires	
-Planned for System MR 4.0		DOCUMENTATION CHANGES	
-Fixes Bug Number(s) _____		Document	Specify One or More
-Documented in MTB _____			
-User/Operations-visible			
Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		MPM (Vol, Sect.)	
-Incompatible change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no		PLMS (AN #)	
-Performance: <input checked="" type="checkbox"/> Better <input type="checkbox"/> Same <input type="checkbox"/> Worse		MOSN (Sect.)	
-Replaces MCR _____		MPAM (Sect.)	
		MSAM (Sect.)	
		Info Segs	
		Other (Name)	
		None (Reason)	
Objections/Comments: lib			

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

SUMMARY:

Install two segments in >ldd>exl>o, new_fortran and bound_fort_. The second already is available in >unb, at a user's request. This series of installations would allow users to have access to a more up-to-date version of the compiler as well as provide greater exposure for this new compiler. The segment new_fortran is not expected in >unb before MR5.0.

REASONS:

Greater exposure for the new compiler.

IMPLICATIONS:

Users can begin to use this much improved compiler before MR5.0

TITLE: Fix mail error message		STATUS	DATE
AUTHOR: S. Herbst		Written	7/1/76
		Status	A 07/13/76
		Expires	01/13/77
-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 5.0		<input type="checkbox"/> Lib. Maint. Tools	
-Fixes Bug Number(s)		<input type="checkbox"/> Sys. Anal. Tools	
-Documented in MTB		<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 checked="" 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 checked="" type="checkbox"/> User Ccmd/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) doc OK	

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

Summary:

Fix bug in mail causing the error message,

"No mailbox for Person.Project",

when the recipient's mailbox exists but cannot be initiated for access reasons.

TITLE: Fix bug in where active function		STATUS	DATE
AUTHOR: S. Herbst		Written	7/6/76
-Coded in: <input checked="" type="checkbox"/> P/L/I <input type="checkbox"/> AIM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Status	A 07/13/76
		Expires	01/13/77
-Planned for System MR <u>5.0</u>		DOCUMENTATION CHANGES	
-Fixes Bug Number(s) _____		Document	Specify One or More
-Documented in MTB _____		MPM (Vol, Sect.)	
-User/Operations-visible 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 <input type="checkbox"/> Worse		MPAM (Sect.)	
-Replaces MCR _____		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 bug causing the where active function to return an extra blank at the end of its return string. This extra blank prevents concatenation.

TITLE: Condense output of where and get_quota		STATUS	DATE
AUTHOR: S. Herbst JWG		Written	07/06/76
<input checked="" type="checkbox"/> Coded in: <input type="checkbox"/> PL/I <input type="checkbox"/> AIM <input type="checkbox"/> other- explain in DETAILED PROPOSAL <input type="checkbox"/> Planned for System MR 5.0 <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 Performance: <input type="checkbox"/> Better <input checked="" type="checkbox"/> Same <input type="checkbox"/> Worse <input type="checkbox"/> Replaces MCR _____		Status	A 07/13/76
		Expires	01/13/77
		DOCUMENTATION CHANGES	
Category (Check One)		Document	Specify One or More
<input type="checkbox"/> Lib. Maint. Tools		MPM (Vol, Sect.)	
<input type="checkbox"/> Sys. Anal. Tools		PLMS (AN #)	
<input type="checkbox"/> Sys. Prog. Tools		MOSN (Sect.)	
<input type="checkbox"/> 355		MPAM (Sect.)	
<input type="checkbox"/> BOS		MSAM (Sect.)	
<input type="checkbox"/> Salvager		Info Segs	
<input type="checkbox"/> Ring Zero		Other (Name)	
<input type="checkbox"/> Ring One		None (Reason)	doc ok
<input type="checkbox"/> SysDaemon/Admin.			
<input type="checkbox"/> Runtime			
<input checked="" type="checkbox"/> User Cmmd/Subr.			

Objections/Comments:

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

SUMMARY:

Change the where command (without -all) and the get_quota command to eliminate the extra blank lines from their output.

REASON:

Other commands do not print unnecessary blank lines.

Currently: where foo

```
>udd>foo_dir>foo
```

```
r 1029 0.120 3.866 86
```

Proposed: where foo

```
>udd>foo_dir>foo
```

```
r 1029 0.120 3.866 86
```

TITLE: Fix bug in start command		STATUS	DATE
AUTHOR: S. Herbst		Written	07/06/76
-Coded in: <input checked="" type="checkbox"/> IPL/I <input type="checkbox"/> ALM <input type="checkbox"/> other- explain in DETAILED PROPOSAL		Status	A 07/13/76
-Planned for System MR 5.0		Expires	01/13/77
-Fixes Bug Number(s)		DOCUMENTATION CHANGES	
-Documented in MTB		Document	Specify One or More
-User/Operations-visible		MPM (Vol, Sect.)	
Interface change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no		PLMS (AN #)	
-Incompatible change? <input checked="" type="checkbox"/> yes <input 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) doc ok	

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

SUMMARY:

Fix the start command to just return after printing the message:

"start" ignored.

REASON:

Currently, it returns to the nearest listener level. This is the wrong thing to do if start was invoked from inside another program, for example:

```
qx
estart
"start" ignored.
(user is at command level.)
```

TITLE: Remove the decoding of version I operator names		STATUS	DATE
AUTHOR: S. Barr TVV		Written	7/7/76
-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 <u>5.0</u> -Fixes Bug Number(s) _____ -Documented in MTB _____ -User/Operations-visible Interface change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no -Incompatible change? <input checked="" type="checkbox"/> yes <input type="checkbox"/> no -Performance: <input type="checkbox"/> Better <input type="checkbox"/> Same <input type="checkbox"/> Worse -Replaces MCR _____		Status	A 07/13/76
		Expires	01/13/77
		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 type="checkbox"/> User Cmmnd/Subr.		Document	Specify One or More
Objections/Comments:		MPM (Vol, Sect.)	
		PLMS (AN #) AN79	
		MOSN (Sect.)	
		MPAM (Sect.)	
		MSAM (Sect.)	
		Info Segs	
		Other (Name)	
		None (Reason)	

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

Reasons:

The data base for version I operator names is kept in the author maintained library which is not shipped to all Multics sites.

Proposal:

Remove operator_names.
 Change find_operator_name_ to ignore requests for version I operator names. This would be the same result as if the instruction were not an operator call.

Implications:

This is an incompatible change.

The following procedures would work differently when they encounter a version I operator.

(callers of find_operator_name_)

```
interpret_ptr_
get_ppr_
```

(callers of the above procedures)

```
dump_machine_cond_
list_frame_args_
print_arg_list_
list_onunits_
```

Internal Interface
 Administrative/User Ring
 10/19/73
 System: 20.9

find_operator_name_

This procedure is given a text reference to one of the PL/I operator segments and returns the name of the operator referenced. Entry operators as well as ordinary operators are handled.

```
declare find_operator_name_ entry (char(*) aligned, ptr,
char(32) aligned);
```

```
call find_operator_name_ (op_seg_name, callp, op_name);
```

op_seg_name is the name of the operator segment being referenced (either `pl1_operators_` or ~~`pl1_operators_`~~). (Input) *rule)*

callp is a pointer to an instruction referencing the operator segment specified by `op_seg_name`. (Input)

op_name is the name of the operator being referenced. (Output)

TITLE: Fix bugs in trace_stack	STATUS	DATE
AUTHOR: S. Barr	TVV	
	Written	7/7/76
	Status	A 07/13/76
	Expires	07/13/77

<input checked="" type="checkbox"/> PL/I <input type="checkbox"/> ALM <input type="checkbox"/> other explain in DETAILED PROPOSAL -Planned for System MR 5.0 -Fixes Bug Number(s) ---- -Documented in MTB ---- -User/Operations-visible Interface change? <input type="checkbox"/> yes <input checked="" type="checkbox"/> no -Incompatible change? <input checked="" 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 Cmmnd/Subr.	DOCUMENTATION CHANGES Document Specify One or More <input type="checkbox"/> MPM (Vol, Sect.) <input type="checkbox"/> PLMS (AN #) <input type="checkbox"/> MOSN (Sect.) <input type="checkbox"/> MPAM (Sect.) <input type="checkbox"/> MSAM (Sect.)
---	---	---

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

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

Summary:

1. Correct interpret_ptr_ to prevent a fault while interpreting a bad pointer.
2. Correct print_source_line_ to recognize the string "PL/I" as version II PL/I.
3. Remove the printing of version I operator names.

Reasons:

- 1,2. These are simple bug fixes as the code exists to handle these cases.
3. The data base for version I operator names is kept in the author maintained library which is not shipped to all Multics sites.

Proposal:

3. Version I operator names will not be recognized and the transfer instruction will be printed instead.