To: Distribution

From: Joan Scott

Date: 30 June 76

Subject: Multics Change Requests

Enclosed are the Multics Change Requests which were approved from 16 June 76 to 30 June 76.
**TITLE:** Update `ol_dump` for the New Storage System (NSS)  
**AUTHOR:** J. Bush

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>System MR 4.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bug Number(s)</td>
<td></td>
<td>355</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Documented in MIB</td>
<td></td>
<td>BOS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-User/Operations-visible</td>
<td></td>
<td>Salvager</td>
<td>MPM (Vol, Sect.)</td>
<td></td>
</tr>
<tr>
<td>Interface change? yes</td>
<td>no</td>
<td>Ring Zero</td>
<td>PLMS (AN #)</td>
<td>53</td>
</tr>
<tr>
<td>Performance: Better</td>
<td>Same</td>
<td>Ring One</td>
<td>MOSN (Sect.)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SysDaemon/Admin.</td>
<td>MPAM (Sect.)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Runtime</td>
<td>MSAM (Sect.)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>User Cmd/Subr.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Objections/Comments:**  
Rewrite in more descriptive form

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

**SUMMARY:**  
Make `ol_dump` compatible with the New Storage System (NSS).

**REASON:**  
Several system include files have been changed, making the current version of `ol_dump` incompatible.

**IMPLICATIONS:**  
The updated version of `ol_dump` will be more helpful in analyzing NSS dumps.
**SUMMARY:** Fix the plus active function to accept a single argument and return the value of that argument:

\[
\text{[plus 2]} \quad \text{returns "2".}
\]

**REASON:** Using \&f in an exec_com, a variable number of arguments can be passed. The plus operation is trivially defined for one argument, but currently says:

"plus: Wrong number of arguments supplied."
LOGICAL ACTIVE FUNCTIONS.

The logical active functions described below return a character string value of either true or false. They are intended to be used with the &if control statement of the exec_com command.

In the descriptions of logical active functions, the term decimal number is used. This includes integers (such as 5 and 1024), real numbers (such as 1.37 and .5), and floating point numbers (such as -1.3e+4 and 12.556e+0).

Name: and

Usage

[and args]

where args are character strings that must have one of the values true or false (if not, an error diagnostic is issued and the value is undefined). If all the args = true, then true is returned; otherwise, false is returned.

[If no args are specified, the value true (the identity element for the and operation) is returned.]
Name: **not**

**Usage**

`[not A]`

where A is a character string. If A = true, then false is returned. If A = false, then true is returned; otherwise, an error diagnostic is issued.

Name: **or**

**Usage**

`[or args]`

where args are character strings that must have one of the values true or false, (if not, an error diagnostic is issued and the value is undefined). If any arg = true, then true is returned; otherwise false is returned.

**Example**

The following example illustrates the use of one of the active functions discussed in this description. It involves the use of the `if` control statement of the `exec_com` command. (See the description of the `exec_com` command.)

```
&if [equal [wd] [home_dir]]
&then &goto elsewhere
&else change_wdir [home_dir]
```

This example compares the pathname of the working directory with the pathname of the home directory, and if they are not the same, changes the working directory to be the home directory.

**ARITHMETIC ACTIVE FUNCTIONS**

The following active functions all perform some arithmetic operation on their arguments and return the character string representation of the result.

In the descriptions of arithmetic active functions, the term decimal number is used. This includes integers (such as 5 and 1024), real numbers (such as 1.37 and .5), and floating point numbers (such as -1.3e+4 and 12.556e+0).
Name: minus

Usage

[minus A B]

where A and B are character strings representing decimal numbers. The value returned is \( A - B \).

- If B is not specified, the value returned is \( -A \).
- If A is not specified, the value returned is zero.

Name: mod

Usage

[mod A B]

where A and B are character strings representing decimal numbers. The value returned is A modulo B.

Name: plus

Usage

[plus args]

where args are character strings representing decimal numbers. The value returned is the sum of args. If no args are specified, the value returned is zero.

Name: quotient

Usage

[quotient A B]

where A and B are character strings representing decimal numbers. The value returned is \( A / B \).

Name: times

Usage

[times A B]

where A and B are character strings representing decimal numbers. The value returned is \( A * B \).

- If B is not specified, the value returned is \( A \).
- If A is also not specified, the value returned is one.
**MULTICS CHANGE REQUEST**

**TITLE:** Create phcs_$terminate_noname  
**AUTHOR:** Bernard Greenberg

**Planned for System:** MR 4.0  
**Fixes Bug Number(s):** not applicable  
**Documented in MTB:** not applicable  
**Incompatible Change:** no  
**User/Operations-visible Interface Change:** no  
**Coded in:** (l)PL/I  
**Performance:** ( )better (l)same ( )worse

**DOCUMENTATION CHANGES (specify one or more)**  
<table>
<thead>
<tr>
<th>MPM (vol,sect)</th>
<th>MPAM (sect)</th>
<th>MPAM (sect)</th>
<th>MSAM (sect)</th>
<th>PLMs (AN#)</th>
<th>61</th>
<th>Info Segs</th>
<th>Other</th>
</tr>
</thead>
</table>

**CATEGOR( check one)**  

**EXPIRES:** 12/03/76

**CATEGORY (check one)**  

**OBJECTIONS/COMMENTS:**

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

**SUMMARY:** Create a gate to the terminate primitive to terminate a segment at zero validation level.

**REASONS:** Outer-ring tools that initiate directories have no way of cleaning up. In the case of tools that walk many directories, KST overflow begins to be a problem.

**IMPLICATIONS:** Ability to list VTOC's without new_proc'ing every few hundred segments, etc.

**DETAILED PROPOSAL:** Implement phcs_$terminate_seg, phcs_$terminate_noname, identical to hcs_counterparts, but operating at a validation level of 0.
SUMMARY:
Fix tty.pl1 to stop looking for an ASCII new line character when reading in raw input mode.

REASONS:
So v2apl can read from 2741s.

IMPLICATIONS:
v2apl will work again. Reading will conform to documentation again.

DETAILED PROPOSAL:
Replace tty_ (submitted as emergency fix to MIT & Phoenix).
## Multics Change Request

<table>
<thead>
<tr>
<th>Title:</th>
<th>Install edit_data_model and edit_data_submodel commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author:</td>
<td>R. Luebke and R. Lackey</td>
</tr>
<tr>
<td>Status</td>
<td>Written 06/07/76</td>
</tr>
<tr>
<td>Date</td>
<td>Expires 12/07/76</td>
</tr>
</tbody>
</table>

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

**DOCUMENTATION CHANGES**
- Specify One or More
  - MFP (Vol, Sect.)
  - PLMS (AN #)
  - MOSN (Sect.)
  - MPAM (Sect.)
  - MSAM (Sect.)

**Document:***
- Unbundled

**Objections/Comments:**

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

---

**SUMMARY:**

Install edit_data_model and edit_data_submodel commands in the unbundled library along with the rest of the Multics Relational Data Store (MRDS).

**REASON:**

The dmd_ and dsmd_ subroutine interfaces to MRDS are intended as internal interfaces and not as user level interfaces, as they require knowledge of Multics descriptor formats. These commands allow definition of a data base in terms of pl/1 data types.
NAME: Edit_data_model, edm

The edm editor can be used to create or examine a data model for the Multics Relational Data Store (MRDS). The edm editor may not be called recursively.

Usage

ed

This command invokes the editor and puts it in edit mode ready to respond to user requests.

Requests

Editor requests are listed below. Either the long name or the short name may be used.

<table>
<thead>
<tr>
<th>Long_name</th>
<th>Short_name</th>
</tr>
</thead>
<tbody>
<tr>
<td>create</td>
<td>cr</td>
</tr>
<tr>
<td>close</td>
<td>cl</td>
</tr>
<tr>
<td>domain</td>
<td>dom</td>
</tr>
<tr>
<td>execute</td>
<td>e</td>
</tr>
<tr>
<td>inspect</td>
<td>i</td>
</tr>
<tr>
<td>list_domains</td>
<td>ld</td>
</tr>
<tr>
<td>list_relations</td>
<td>lr</td>
</tr>
<tr>
<td>print</td>
<td>pr</td>
</tr>
<tr>
<td>relation</td>
<td>rel</td>
</tr>
<tr>
<td>quit</td>
<td>q</td>
</tr>
</tbody>
</table>

A description of each request follows.

create
Format: create path
Purpose: To create the data base with the pathname specified and open the data model for creation. All editor requests from this point until the next close request will refer to this data base.

close
Format: close
Purpose: To close the currently open data model.
Note: After a create or inspect request, a close request must be issued before a quit request will be honored.

**Domain**

Format: `domain name declaration`

Purpose: To define a domain by name and attach a type definition to it prior to defining relations using the domain definition.

Name: is the name of the domain to be defined. It must be 1 to 32 characters long and may be composed of alphanumeric characters and underscore (_).

declaration: is a data type declaration using PL/1 syntax.

Examples: `domain employee_number fixed decimal (6)`
`domain last_name char (20) aligned`

**Execute**

Format: `execute command_line`

Purpose: To allow calling any Multics command from within the editor.

**Inspect**

Format: `inspect path`

Purpose: To open the data model of a specified data base for inspection purposes. All editor requests from this point until the next close request will refer to this data model.

Note: Only the print, execute, and list_relations requests may be used while an inspect opening is in effect.

**List_domains**

Format: `list_domains`

Purpose: To list the currently defined domains and their type definitions.
### List_relations

**Format:**

```
list_relations
```

**Purpose:**

To list the names of all the relations in the current data model.

### Print

**Format:**

```
print relation_name -control_arg-
```

**Purpose:**

To print information about relations in the data model.

*relation_name*

is the name of the relation about which information is to be printed. If relation_name is "all" then information about all the relations is printed.

The following control argument may be specified:

- `-long`, `-lg`

prints bit offset and length information about each attribute in addition to the information normally printed.

### Relation

**Format:**

```
relation relation_expression
```

**Purpose:**

To define a relation in the data model.

*relation_expression*

is a relation expression as defined in the section "Relation Expressions" which defines the desired relation.

**Note:**

All domain names included in the relation expression must have been previously defined by a domain request.

### Quit

**Format:**

```
quit
```

**Purpose:**

To exit the editor and return to Multics command level.

### Quit
Format:

Purpose: To print "edit_data_model" on user_output to indicate that control is within the editor.

Data Type Declarations

The data type of a domain is declared using a subset of the PL/1 declare statement syntax. The following PL/1 declaration attributes are supported:

- aligned
- binary
- bit
- character
- complex
- decimal
- fixed
- float
- nonvarying
- precision
- real
- unaligned
- varyng
- var

A semicolon must not follow the end of the type declaration.

Relation Expressions

When defining a relation it is necessary to specify which attributes are to be used as components of the primary key (i.e., key attributes). Each tuple must be identifiable by some non-null primary key value which must be unique for any one relation.

The expression for defining a relation and its attributes resembles the format used in most of the literature (i.e., relation_name (attribute_one attribute_two ... ) where the underscore denotes a key attribute). In the following syntax the underscore is replaced by a following asterisk. It is also possible to rename data model attributes so that they may differ from the corresponding data model attribute names.

The relation definition expression provides a method for defining a relation and its attributes.
Formal Syntax:

<relation-definition-expression> ::= <relation-expression> <attribute-expression>

<relation-expression> ::= <identifier>

<attribute-expression> ::= (<attribute-definition> [<attribute-definition> ... ])

<attribute-definition> ::= <attribute> [<key-indicator>]

<key-indicator> ::= ?

<attribute> ::= <domain> [<role>]

<domain> ::= <identifier>

<role> ::= <identifier>

<identifier> ::= <letter>[<letter>]<digit> ...  

<letter> and <digit> are as defined in the PL/I Language Manual. 
Attributes must be no longer than 32 characters.
The edsm editor can be used to create and edit a data submodel (.dsm) associated with the Multics Relational Data Store (MRDS) subsystem. The edsm editor cannot be called recursively.

All data submodels will have the suffix .dsm on the segment names, however, user references to data submodel need not include the .dsm suffix.

**Usage**

```
edsm -optional dsm pathname -
```

where:

- optional dsm pathname - is an optional argument and, if present, specifies the pathname of a previously created data submodel that the user wishes to operate upon. The editor will, if necessary, concatenate the suffix .dsm to the pathname to obtain the complete data submodel pathname.

Once the edsm editor is invoked, the user can immediately begin to issue edsm requests from his terminal. Requests fall into one of four general categories: read and create requests, write requests, modify and display requests and special editor commands. The edsm editing functions are not performed directly on the target data submodel but in a temporary copy of the target data submodel. The temporary copy is made permanent when a write request is executed.

The edsm editor is designed to allow the user to abort any editor command by pressing the interrupt button on his terminal. This will cause the Multics system to print QUIT on the user's terminal and establish the user at Multics command level. To return to the edsm editor, the user should issue the Multics program_interrupt (pi) command which will place the user back to the edsm editor command level.
To create a new DSM submodel, a user might perform the following steps:

1. **Invoke the DSM editor by typing the `multics command 'edsm'.**

2. **Request that a new DSM be created by typing the 'create' command followed by the relative pathname where the DSM will be permanently stored.**

3. **Specify the pathname of the database with which this data submodel is to be associated by typing 'database_path' followed by the absolute pathname of the database.**

4. **Define a relation for the data submodel by typing the 'relation' command followed by the relation definition expression.**

5. **Inspect the contents of the DSM by typing the 'print' command.**

6. **Write the contents of the temporary data submodel to permanent storage by typing the 'write' command.**

7. **Exit from the editor using the 'quit' command.**
To edit an existing data submodel, a user might perform the following steps:

1. Invoke the edsm editor by typing the Multics command 'edsm'.
2. Read the existing data submodel on to temporary storage by typing 'read' followed by the relative pathname of the existing dsm.
3. Editing the data submodel using the 'delete', 'relation' or 'database_path' commands.
4. Using the 'write' request, write the contents of the temporary data submodel either into the original segment or, into a segment specified by a pathname.
5. Exit from the editor using the 'quit' request.

The user can create and edit any number of data submodels with a single invocation of the editor, providing the contents of the temporary dsm is written prior to making a new read or create request.

Note

While most users interact with the edsm editor through a terminal, the editor is designed to accept input through the user_input I/O switch and transmit output through the user_output switch. These switches can be controlled (using the iox_subroutines described in the VPM Subroutines) to interface with other devices/files in addition to the user's terminal. For convenience, the edsm editor description assumes that the user's input/output device is a terminal.

All edsm editor error messages will be directed to the error_output switch.
LIST OF EDITOR REQUESTS

In the list given below, edsm editor requests are divided into four categories: read and create requests, edit and display requests, write requests and special requests. The create, read, write, edit and display requests are sufficient to create and edit data submodels. The special requests allow the user to determine the status of the temporary dsm, to execute Multics commands without leaving the editor and to list the long and brief forms of the editor commands.

All edsm editor requests must be separated from their arguments with at least one blank or tabulation. Only one editor request is allowed per input line. The relation request is treated somewhat differently in that once the 'relation' command has been given, any further text on that input line and subsequent input lines will be treated as the relation expression. The first right parenthesis encountered following the relation command will terminate the relation expression. Return to the editor command level occurs when either this terminator is found or an invalid relation expression format is detected.

Except for create, those request that reference a current data submodel when one has not been established (i.e., a prior read or create request has not been executed) the editor will issue the message "No data submodel open."

In the following description the characters given in parentheses are the brief forms of the editor commands or request.

CREATE AND READ REQUESTS

create (cr) <ism pathname> creates a temporary data submodel. The pathname is a relative pathname of the segment where the data submodel will be stored with a subsequent write request. If the segment name does not end with the suffix dsm it will be appended by the editor to form the complete pathname. If a segment already exist by the name specified in dsm pathname, the create request will not be allowed. Also the create request will not be allowed if a current temporary dsm exists and has been modified since it was last written.
**READ REQUESTS**

`read (r) <dsm pathname>` reads an existing data submodel residing in the segment specified by the relative dsm pathname. If the pathname does not contain the suffix .dsm the editor will append the .dsm suffix prior to attempting to read the dsm. Checks are performed to insure that the object being read is indeed a data submodel. The `read` command will not be allowed if a current temporary dsm exists and has been modified since it was last written.

**WRITE REQUESTS**

`write (w) <optional pathname>` copies the temporary data submodel into the user's specified segment. If no pathname is supplied with the write request the pathname that was supplied with the last prior read or create request is used. If a pathname is supplied with the write request it is used for the write operation. If the segment being written does not exist, it will be created. If it does exist, its contents will be overwritten with contents of the temporary data submodel. If no database pathname has been specified a warning message will be issued by the editor but the write will still be completed. If no temporary data submodel exists (i.e. a prior read or create request had not been done) the editor will issue the message "No data submodel open".

**EDIT AND DISPLAY REQUESTS**

`database_path (dp)` `<database pathname>` adds or modifies the database pathname to which the current temporary data submodel is associated. The database pathname must be an absolute pathname and is stored in the header of the dsm.
delete (dl) <relation name> deletes the relation specified by the relation name from the current temporary dsm. If the relation does not exist in the dsm the message "Relation rrrrr not found" where rrrrr is the relation name supplied with the delete request.

relation (rel) <relation expression> adds a new relation to the current temporary dsm. If a relation already exists by the name supplied with the relation request, the message "Relation rrrrr has been previously defined." where rrrrr is the name of the relation supplied in the relation expression. The relation expression may be continued on more than one line. Once the relation request has been issued, the input expected is a relation expression and subsequent input will be treated as part of the relation expression until the right parenthesis terminating character of the expression is found or a relation expression parsing error is encountered. If a relation parsing error is encountered an explanatory error message will be issued and control returned to the editor command level.

A detailed description of the relation expression is supplied later in the document.

list_relations (lr) Prints an alphabetical list of the relations in the current temporary data submodel preceded by a header specifying the submodel pathname associated with the current dsm. If no relation exists in the dsm, the header is printed followed by the message "No relations in data submodel."
print (pr) <optional arg> if no arguments are supplied
with the print request an alphabetically
ordered list of the existing DSM
relations and their associated attribute
names is printed preceded the current
submodel pathname and the current DSM
header information.

If a relation name is supplied as the
optional argument, only the specified
relation and its associated attribute
names is printed. The error message
"RRRRR not found." is printed if
the relation does not exist, where RRRRR
is the relation name supplied with the
print request.

If the optional argument is -header or
-shared, only the header information for the
current temporary DSM is printed.

SPECIAL REQUESTS

commands (com) <optional arg> if no optional arguments are
supplied a list of the long form of the
edsm editor commands is printed.

If the optional argument -brief or -S is
supplied a list of the brief or short
form of the edsm editor commands is
printed.

not (.)
prints the name 'edsm' on the user's
terminal. Its purpose is to allow the
user to determine if he is currently in
the edsm editor.

execute (e)
passes the remainder of the request line
to the multics command processor (i.e.,
escapes to to execute other Multics
commands).

quit (q)
when spelled with a capital, exits from
the editor returning the user to Multics
command level. The current temporary
DSM, if any, is deleted.
when spelled with lower case, will exit from the editor only if the temporary data submodel, if any, has not been modified since it was last written. If the current temporary DSM has been modified since it was last written the message "The data submodel xxxxx has been modified since last write, quit rejected." will be printed on the user's terminal.

print the pathname of the current data submodel. If the DSM has been modified since it was last written the pathname will be prefixed with [modified]. If no DSM is open the message "No data submodel open." will be printed on the user's terminal.
The following is a table of the edsm editor commands, and a brief description of their function. The brief form for each command is included in parentheses.

- **commands (com)**: prints a list of the edsm editor commands.
- **create (cr)**: creates a temporary data submodel.
- **database_path (dp)**: adds or modifies the database pathname to which the current temporary data submodel is associated.
- **delete (dl)**: deletes a relation from the temporary data submodel.
- **not (.)**: print the name "edsm" on the users terminal.
- **execute (e)**: escapes to execute Multics command level commands.
- **list_relations (lr)**: prints a list of the names of the relation in the current data submodel.
- **print (pr)**: prints the relations in the current data submodel.
- **Quit (\)**: exits from the editor without any modification checking.
- **quit (a)**: exits from the editor with modification checking.
- **read (r)**: reads an existing data submodel into temporary storage making it the current data submodel.
- **relation (rel)**: adds a new relation to the current data submodel.
- **status (st)**: prints the pathname of the current data submodel.
- **write (w)**: copies the current temporary data submodel into user storage.
Data_submodel_relation_expressions

The function of a data submodel relation expression is to associate the user's view of a database with the actual database description (i.e., the data model). This association is accomplished by equating the user's name for a relation term to the data model's name for the same term.

The major components of a data submodel relation expression are the relation names and the attribute names. The general form of a data submodel relation expression is as follows:

relation name (attribute names)

The first attribute name must be separated from the relation name with a left parenthesis and the last attribute name must be followed with the right parenthesis expression terminator. Attribute names must be separated with at least one blank except the case where attribute names are equated, see below.

If the user's view of a relation name differs from that specified in the data model, the user's relation name is equated to the data model specification name with an equal sign in the following manner:

user_rel_name = data_model_rel_name (attribute names)

If only one relation name is supplied in the dsm relation expression, it is assumed that the user does not require a different relation name from that specified in the data submodel (i.e., the data submodel and data model relation names are the same).

The same is true for the user's view of attribute names:

relation_name (user_attr_name_1 = dm_attr_name_1 .......

If the user's attribute name is the same as the data model attribute name, only a single name is required. The following data submodel expressions are equivalent:

\[ x \quad a=b \quad c \quad u \]
\[ x=x \quad a=b \quad c=c \quad w=d \]

If an invalid data submodel relation expression is detected, the relation definition will be aborted, an explanatory error message will be issued and control returned to the editor command level.
The following example creates a data submodel by the name location.dsm and is built using the data model relation below:

```
phone_book (name phone mail_station room plug dept_code)
```

Terminal input lines are prefixed with the symbols "!".

! dsm
! or location
! or -m:

```
version_number = 1
date_time_generated = 03/05/76 1109,1 mst Fri
database_pathname = ••••••••NO-DATABASE-PATHNAME••••••
number_of_relations = 0
```

! run "dsmktst2_telephone_info"

! rel
! location = phone_book
! (building = alig
! office=room
! person=none)

! list
! Dsm does not recognize the command 'list'
!
!
! list

Data submodel relation names for >udd>m>rdl>asm_i>location.dsm

location

! list "modified" dsmktst2_telephone_info.dsm>location.dsm"
Data Submodel: udd>m>rdl>osm_i>location.dsm

<table>
<thead>
<tr>
<th>Version_number</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date_time_generated</td>
<td>03/05/76 1109.5 mst Fri</td>
</tr>
<tr>
<td>Database pathname</td>
<td>udd&gt;m&gt;test&gt;telephone_info</td>
</tr>
<tr>
<td>Number of relations</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Submodel Relation Name</th>
<th>Model Relation Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>location</td>
<td>phone_book</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Submodel Attributes</th>
<th>Model Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>building</td>
<td>bldg</td>
</tr>
<tr>
<td>office</td>
<td>room</td>
</tr>
<tr>
<td>person</td>
<td>name</td>
</tr>
</tbody>
</table>
**Title:** Fix stringsize signalling on zero-length moves.

**Author:** Bernard Greenberg

<table>
<thead>
<tr>
<th>Planned for System:</th>
<th>MR 4.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixes Bug Number(s):</td>
<td>not applicable</td>
</tr>
<tr>
<td>Documented in MTB:</td>
<td>not applicable</td>
</tr>
<tr>
<td>Incompatible Change:</td>
<td>no</td>
</tr>
<tr>
<td>User/Operations-visible Interface Change:</td>
<td>no</td>
</tr>
</tbody>
</table>

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

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

- MPM (vol, sect)
- MPAM (sect)
- MOSN (sect)
- MSAN (sect)
- PLMs (AN#) 78
- Info Segs
- Other

**OBJECTIONS/COMMENTS:**

**SUMMARY:** EIS moves into zero length targets are currently simulated by a hardcore procedure to be an effective no-op. However, in the case where the "enablefault" bit is on in the instruction (compiled by PL/I if stringsize is enabled), an illegal-procedure fault is signalled. Change the FIM to signal stringsize instead in this case.

**REASON:** Adherence to PL/I specification vs. arbitrary behavior.

**IMPLICATIONS:** This is all unnecessary when the hardware fix for zero-target-length moves is propagated.

**DETAILED PROPOSAL:** Implement a new return protocol between the FIM and hardcore fault handlers, whereby a certain bit in the machine conditions (mc.resisignal) will cause the FIM to signal a condition of the fault handler's choice instead of the FIM's original choice. Change illegal_procedure to use this to cause stringsize to be signalled instead of illegal_procedure. Change seg_fault, recently changed to signal out_of_bounds in certain cases via an ad-hoc method, to use this general method instead, to signal out_of_bounds instead of seg_fault_error. Do not signal anything if both source and target string length are zero.
SUMMARY: Miscellaneous errors and oversights in virtual time accounting can cause the traffic controller's cumulated eligibility time to go negative. This has disastrous consequences, putting the system in a preemption loop. Cause the traffic controller to ignore negative cumulations of eligibility time.

REASONS: More reliable operation. Contributes to lost timers and system loops.

IMPLICATIONS: n/a. Negative-te processes will go through a nugatory eligibility loss.

THOUGHT: The many various virtual-time accounting bugs are being found and fixed, and 4.1 has a better policy on this altogether. This MCR fixes what is now a serious bug.
**SUMMARY:**
Fix bug that ignores all but the first pair of disk drives on a UDSK card.

**REASONS:**
Fix needed to allow non-consecutive disk drives to be defined as user disk drives.

**IMPLICATIONS:**
Non-consecutive drives may be used as user disk drives.

**DETAILS:**
Fix bug in `rcp_init`. 
TITLE: Fix bug in link_unsnap_

AUTHOR: M. Weaver

- Code in: [X] PL/I [ ] ALM [ ] Other explain in DETAILED PROPOSAL
- Planned for System MR 4.0
- Fixes Bug Number(s)
- Documented in MTB 355
- User/Operations-visible
  - Interface change? [ ] yes [X] no
  - Incompatible change? [ ] yes [X] no
  - Performance: [ ] Better [X] Same [ ] Worse
- Replaces MCR

Objections/Comments:

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

SUMMARY:

Fix link_unsnap_ to not unsnap all links.

REASONS:

When the segment being terminated does not have separate static, all links in the process are unsnapped. This significantly degrades system performance.

Emergency fix - already installed.
The check_mdcs command calls an existing ring1 procedure that validates the format of the MDCS and the uidpath names it contains.

**Detailed proposal**

**name**: check_mdcs

The check_mdcs command checks the MDCS for a given volume for valid format, and for invalid UID pathnames. Any errors found are reported via the syserr log, and if possible, corrected.

**Usage**

```
check_mdcs volume
```

where volume is the name of a storage system volume.

**Notes**

Access to the gate mdc_priv_ is required to use this command.
TITLE: Turn off ioi error logging for T&D attachments
AUTHOR: Larry Johnson

Planned for System: MR 4.1
Fixes Bug Number(s): not applicable
Incompatible Change: no
User/Operations-visible Interface Change: no
Coded in: ( )PL/I ( )ALM ( )other-see below
Performance: ( )better ( )same ( )worse

DOCUMENTATION CHANGES (specify one or more)
MPM (vol,sect) MPAM (sect)
MOUSN (sect) MSAM (sect)
PLMs (AN#)
Info Segs
Other
None (reason)

OBJECTIONS/COMMENTS:

Summary

Modify ioi_ to not log errors which occur during a privileged attachment.

Reasons

Privileged attachments are used only by T&D programs. Errors generated by these programs would not be of interest to HEALS. In addition, one T&D program managed to generate over 30000 sysserr messages in an hour attempting to diagnose a printer problem.
TITLE: New version of BLAST command

AUTHOR: Robert S. Coren

Planned for System: MR 5.0
Fixes Bug Number(s): 393
Documented in MTB: not applicable
Incompatible Change: no
User/Operations-visible Interface Change: no

Coded in: ( ) PL/I ( ) ALM ( ) other—see below
Performance: ( ) better ( ) same ( ) worse

DOCUMENTATION CHANGES (specify one or more)
MPM (vol, sect) MPAM (sect)
MOSN (sect) MSAM (sect)
PLMs (AN#) 85
Info Seqs
Other

OBSERVATIONS/COMMENTS:

HEADINGS ARE: SUMMARY, REASONS, IMPLICATIONS, DETAILED PROPOSAL (optional)

SUMMARY: Change BOS BLAST command to send one message to the 355 (in SCL, EBCDIC, and correspondence) instead of sending individual ones for each configured communications channel. The 355 will then distribute the message only to those lines that are dialed up at the time.

REASONS: Present version often causes 355 to crash while doing BLAST, particularly if many more lines are configured than are actually dialed up.

IMPLICATIONS: Because the 355 cannot distinguish EBCDIC from correspondence terminals, the message will be sent to IBM-type terminals in both codes.

DETAILED PROPOSAL: Invent a new mailbox opcode, "blast", which is sent by BOS to the 355 along with the address of the blast message. The 355 reads the message and then sends to all terminals which have dataset-ready and carrier-detect up.

NOTE: Portions coded in 355map.
TITLE: Interim fix to ld355

AUTHOR: Robert S. Coren

Planned for System: MR 5.0

Fixes Req Number(s): not applicable

Documented in MTR: not applicable

Incompatible Change: yes

User/Operations-visible Interface Change: yes

Code in: ( ) PL/I ( ) ALM ( ) other-see below

Performance: ( ) better ( ) same ( ) worse

DOCUMENTATION CHANGES (specify one or more)

- MPAM (sect)
- MSAM (sect)
- Info Seqs

Other

NONE (reason: ld355 is going away)

OBJECTIONS/COMMENTS:

SUMMARY: Modify BOS ld355 command to look for status in word 6 of mailbox instead of word 1, and to ignore status sent by 355 when initialization program is entered.

REASONS: New 355 init for bootload with Multics sends status when entered and puts status in word 6. Making these changes to ld355 enables us to install the new 355 software (whose core image is made with bind_fnp rather than build355) before 355 bootload from Multics is ready. Once the 355 is booted from Multics, ld355 will no longer be needed.

IMPLICATIONS: New ld355 won't work with old (build355) MCS, nor will new (bind_fnp) MCS work with old ld355.
MULTICS CHANGE REQUEST

TITLE: Fix bug in define_area_ causing random null pointer references.

AUTHOR: Steve Webber

Planned for System: MR 4.0
Fixes Bug Number(s): unreported
Documented in MTR: not applicable
Incompatible Change: no
User/Operations-visible Interface Change: no
Coded in: (X)PL/I ( )ALM ( )other-see below
Performance: ( )better (X)same ( )worse

DOCUMENTATION CHANGES (specify one or more)

XPM (vol,sect) MPAM (sect)
XOSM (sect) HSAM (sect)
PLMs (AND)
Info Secs
Other
None (reason) no change

ORJCTIONS/COMMENTS:

SUMMARY:

Fix bug in define_area_ resulting in null pointer references later on when the area is released.

REASONS:

Bug fix.

DETAILED PROPOSAL:

Change define_area_ to save the name of the program requesting the area even if the area is not extensible. This is needed by subsequent calls to release_temp_segments_ when (if) release_area_ is called on the area. The name is saved in the standard extend_block used for extensible areas and is allocated in the very area.
SUMMARY: Delete the old message facility several months after the old facility is deleted.

REASON: Most users should be converted to the new facility by now.

DETAILED: Delete old_message_facility and change new_message_facility not to print con_mgs messages when printing messages.
TITLE: Delete old mail facility

AUTHOR: S. Herbst

- Coded in: [ ] PL/I [ ] ALM [ ] other
- Planned for System MR 5.0
- Fixes Bug Number(s)
- Documented in MTB
- User/Operations-visible Interface change? [ ] yes [ ] no
- Performance: [ ] Better [X] Same [ ] Worse
- Replaces MCR

Objections/Comments:

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

SUMMARY:

De-install the obsolete old mail facility.

REASON:

New mail has been running for several months and most users have no more use for old mailboxes.

DETAILED PROPOSAL:

Remove the command old_mail.
TITLE: Fix bug in find_command_and full_find_command_

AUTHOR: S. Herbst

STATUS  DATE
Written  5/10/76
Expires  12/28/76

LOCATION OF PROPOSAL
Lib. Maint. Tools
Sys. Anal. Tools
Sys. Prog. Tools

PLANNED FOR SYSTEM
System MR 5.0

DOCUMENTATION CHANGES
Document Specify One or More

USER/OPERATIONS-VISIBLE
Salvager MPM (Vol, Sect.)
BOS Ring Zero PLMS (AN #)
Ring One
MOSN (Sect.)
SysDaemon/Admin.
MPAM (Sect.)
Runtime
MSAM (Sect.)
User Cmd/Subr.

Objections/Comments:
None

SUMMARY:

1. Change find_command_$fc_no_message to call full_find_command_$fc_no_message instead of full_find_command_.

2. Add the entry point name $fc_no_message to full_find_command_$fc_print_message. This entry point suppresses printing of error messages as does find_command_$fc_no_message.

3. Fix bug in full_find_command_ causing the wrong message "segment_u...not found."

when the command name is ":>".

REASONS:

1. Fix bug where find_command_$fc_no_message prints messages.

2. Better mnemonic name for the entry point and analogous to find_command_$fc_no_message.

3. The correct error message in this case is "This operation is not allowed for a directory."
### Fix bug in check_vtoce_branch

**Author:** A. Kobziar  
**Written:** 6/14/76  
**Status:** A  
**Expires:** 12/32/76

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>TVV</th>
<th>STATUS</th>
<th>DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td></td>
<td>Written</td>
<td>6/14/76</td>
</tr>
<tr>
<td>Sys. Maint. Tools</td>
<td></td>
<td>Status</td>
<td>A</td>
</tr>
<tr>
<td>Expire</td>
<td></td>
<td>Expires</td>
<td>12/32/76</td>
</tr>
</tbody>
</table>

#### Document

- **Proposed Changes:**
  - Documented in Lib. Maint. Tools
  - Documented in Sys. Maint. Tools

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>TVV</th>
<th>STATUS</th>
<th>DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td></td>
<td>Written</td>
<td>6/14/76</td>
</tr>
<tr>
<td>Sys. Maint. Tools</td>
<td></td>
<td>Status</td>
<td>A</td>
</tr>
<tr>
<td>Expire</td>
<td></td>
<td>Expires</td>
<td>12/32/76</td>
</tr>
</tbody>
</table>

#### Fixes Bug Number(s)

- Fix bug which may occur when volume salvaging with the `check_dir` option caused by the use of the same automatic variable by internal procedures. The bug causes an execution of ascii data and a crawlout.


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

**SUMMARY:**

Re-implement the PL/I allocate statement, free statement, and empty builtin to generate transfers to `p1l_operators_` rather than subroutine calls.

**REASONS:**

Better performance and only way to get full ANSI implementation of allocate statement and area condition.

**DETAILED PROPOSAL:**

1. Implement the operator interface in `alloc_` and `p1l_operators_` for allocate, free and empty.
2. Fix a bug in `alloc_` that caused allocation not to be retried after return from an area on-unit.
3. Remove a restriction that prevented reevaluation of the in option of an allocate statement during allocate retry after return from an area on-unit.
4. Make changes to `p1l` to generate direct transfers to `p1l_operators_` for allocate, free, and empty.
In the following discussion, error_output is understood to be the Multics data stream attached to the name "error_output".

Although the description of each condition states when the condition occurs, the following conditions may occur anytime during execution of the program:

- underflow
- overflow
- fixedoverflow
- zerodivide
- size
- stringsize
- storage
- area
- error

These conditions occur when the compiled code or any of its supporting subroutines exceed one or more of their limitations or when they detect an error. Execution of a valid program does not normally cause these unexpected conditions to occur.

10.4.1 The Area Condition

Syntax:

<area condition name>::= area

This condition occurs when an <allocate statement> attempts to allocate a generation of a based variable in an area whose size is insufficient to contain the generation, or when an <assignment statement> assigns an area to an area whose size is insufficient to contain the assigned area. If an <on unit> returns to the point where the condition was detected and the condition was signalled by the execution of an <assignment statement>, the program is in error. If the condition was signalled by the execution of an <allocate statement>, the allocation is retried. Unless the <on unit> has freed sufficient storage in the area, the condition will occur again.

The default <on unit> writes a comment on error_output and signals the error condition.

10.4.2 The Conversion Condition

Syntax:

including reevaluation of the <in option> of the <allocation>.

or caused the value of the <in option> to change to an area that had sufficient storage.
**Multics Change Request**

**TITLE:**  Update program to print vtoc man metering info

**AUTHOR:** A. Bensoussan

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**PL/I** [ ] **ALM** [ ] **other** [ ]

**Planned for System MR 4.0**

**Fixes Bug Number(s):**

**Documented in MTB:**

**User/Operations-visible Interface change?**

**Performance:**

**Replaces MCR**

**STATUS DATE**

**Written:** 06/10/76

**Status:** A 06/22/76

**Expires:** 12/22/76

**DOCUMENTATION CHANGES**

<table>
<thead>
<tr>
<th>Document</th>
<th>Specify One or More</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Objections/Comments:**

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

**SUMMARY:**

Modify "vtoc_buffer_meters" to be compatible with the new vtoc manager.
**MULTICS CHANGE REQUEST**

**Title:** Fix verify_lock unlock-failure bug.

**Author:** Bernard Greenberg

**Planned for System:** MR 4.0

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

**Documented in MTB:** not applicable

**Incompatible Change:** no

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

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

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

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

- MPM (vol, sect)
- MUSN (sect)
- PLMS (AN#) 61
- Info Segs
- other

**OBJECTIONS/COMMENTS:**

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

**SUMMARY:** A bug in verify_lock causes it not to unlock directories after invoking the online-salvager on them. Stop this behavior.

**REASONS:** This is a bug.

**IMPLICATIONS:** Fewer notify-timeouts, mylocks, etc.
Ver. 3 741022 MULTICS CHANGE REQUEST

<table>
<thead>
<tr>
<th>TITLE: Fix bug in dprint</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTHOR: VanVleck</td>
</tr>
</tbody>
</table>

| 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: (X)PL/I ( )ALM ( )other-see below |
| Performance: ( )better (X)same ( )worse |

<table>
<thead>
<tr>
<th>DOCUMENTATION CHANGES (specify one or more)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPM (vol,sect)</td>
</tr>
<tr>
<td>MPAM (sect)</td>
</tr>
<tr>
<td>MOSN (sect)</td>
</tr>
<tr>
<td>MSAM (sect)</td>
</tr>
<tr>
<td>PLMs (AN#)</td>
</tr>
<tr>
<td>Info Seqs</td>
</tr>
<tr>
<td>Other</td>
</tr>
<tr>
<td>None (reason) bug fix</td>
</tr>
</tbody>
</table>

| OBJECTIONS/COMMENTS: |

<table>
<thead>
<tr>
<th>STATUS</th>
<th>DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Written:</td>
<td>06/16/76</td>
</tr>
<tr>
<td>Status:</td>
<td>06/23/76</td>
</tr>
<tr>
<td>Expires:</td>
<td>12/16/76</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CATEGORY (check one)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(X)Lib. Maint. Tools</td>
</tr>
<tr>
<td>( )Sys. Anal. Tools</td>
</tr>
<tr>
<td>( )Sys. Prog. Tools</td>
</tr>
<tr>
<td>( )355</td>
</tr>
<tr>
<td>( )BOS</td>
</tr>
<tr>
<td>( )Ring Zero</td>
</tr>
<tr>
<td>( )Ring One</td>
</tr>
<tr>
<td>( )SysDaemon/Admin</td>
</tr>
<tr>
<td>( )Runtime</td>
</tr>
<tr>
<td>(X)User Command/Subr</td>
</tr>
</tbody>
</table>

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

**SUMMARY:**

The entry name is not printed in the dprint error message
dprint: >udd>m>vv> not found. Do you still wish request?
due to a missing ^a. Cause the name to be printed.

**REASONS:**

The message is more helpful if the entry name is printed, especially if
more than one nonexistent segment is requested in a command invocation, or
if dprint is called from an exec_com.
**SUMMARY:** Change transliteration tables in the GCOS Environment to translate the bcd up-arrow to ascii circumflex and the converse as is done in real GCOS. Re-compile procedures using the include file `gcoss_xlate_bcd_ascii_incl.pl1`.

**REASON:** GCOS programs transferred to Multics are aborting because of the difference between the translation rules for GCOS and the current simulator rules.

**IMPLICATIONS:** Bug will be fixed.

**DETAILED PROPOSAL:** Replace:
- `gcoss_xlate_bcd_ascii_incl.pl1`
- `gcoss_cv_bcd_ascii_alm`
- `gcoss_cv_ascii_bcd_alm`
- `gcoss_syspunch.pl1`
- `gcoss_sys_xlate_alm`
- `cv_bcd_to_ascii_alm`
- `cv_bin_to_ascii_pl1`
- `cv_bin_to_bcd_pl1`
- `cv_ascii_to_bcd_pl1`

Re-compile:
- `gcoss_mme_bort_pl1`
- `gcoss_mme_fadd_pl1`
- `gcoss_mme_fcon_pl1`
The program get_system_free_area_ does not set the validation level before creating system_free_. This causes system_free_ to be created with ring brackets of 4,4,4. It should be changed to set the validation level before creating the segment.

Reasons

This is a security problem.
**TITLE:** Change format of syserr log

**AUTHOR:** Larry Johnson

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

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

**CATEGORY (check one):**
- ( )Lib. Maint. Tools
- ( )Sys. Anal. Tools
- ( )Sys. Prog. Tools
- ( )355
- ( )BOS
- ( )Salvager
- (H)Ring Zero

**DOCUMENTATION CHANGES (specify one or more):**
- MPAM (sect)
- MSAM (sect)
- Info Segs
- Other

**None (reason)**

**OBJECTIONS/COMMENTS:**
Requires SRB warning. (Please copy this command onto the yellow form)

---

**Summary:** The syserr log seems to be extremely prone to damage on crashes where ESD fails. This is probably due to the fact that pages of the log are in core and being updated at the time of the crash (logging the reason for the crash) and can't be written out to disk. This may cause threading problems in the log, especially if the new messages cross a page boundary. The log should be reformatted slightly to eliminate this problem.

**Reasons:** Greater reliability of the syserr log, which is coming into greater use because of HEALS.

**Detailed Proposal**
1. Never log a message across a page boundary. If the message will not fit in the current page, skip to the next page.

2. When the log wraps around, move the first message pointer ahead one full page, instead of a message at a time.

3. Increase the syserr header to include a date time modified for each page, and, as each page is filled, a checksum.

4. Modify the system initialization procedure that checks the log to validate the times and checksums for each page. If discrepancies are found, simply unthread the bad page so that outer ring procedures will never see it. The bad page will be reused when the log wraps around to that point again.

**Implications:** Slightly fewer messages will fit in the log. All messages in the ring0 log when this system is installed will be deleted. No changes will be necessary to ring 4 procedures that examine the log.
### Multics Change Request

**Title:** Fix spelling errors in mac_<br>
**Author:** Larry Johnson<br>
**Planned for System:** MR 4.0<br>
**Fixes Bug Number(s):** Not applicable<br>
**Documented in MTB:** Not applicable<br>
**Incompatible Change:** No<br>
**User/Operations-visible Interface Change:** No<br>
**Coded in:** ( ) PL/I ( ) ALM ( ) Other—See Below<br>
**Performance:** ( ) Better ( ) Same ( ) Worse<br>

#### Documentation Changes (specify one or more)

- MPM (vol, sect) MPAM (sect)
- MUSN (sect) MSAM (sect)
- PLSM (AN#)
- Info Segs
- Other

#### Objections/Comments:

---

**Summary**

Two external references, one of an error_table_code, and one of an admin_gate_entry, were misspelled by bound_mail_control in 26-8. These must be corrected. These changes will be included in 26-8a.
Title: Install patch_firmware and dump_firmware tools
Author: Larry Johnson

Planned for System: MR 4.1
Fixes Bug Number(s): not applicable
Documented in MTB: not applicable
Incompatible Change: no
User/Operations-visible Interface Change: no
Coded in: (X)PL/I ( )ALM ( )other-see below
Performance: (X)better ( )same ( )worse

DOCUMENTATION CHANGES (specify one or more)
MPM (vol,sect)  MPAM (sect)
MOSN (sect)  MSAM (sect)
PLMs (AN#)  
Info Segs  
Other Hardware Diagnostic Aids

OBJECTIONS/COMMENTS:
 Resolve third paragraph under "Notes" (patch_firmware writeup).
 New MCR required if new control argument needed.

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

Summary
Install commands that can patch and dump segments containing MPC firmware.

Reasons
It is occasionally necessary to patch firmware. (MR4.0 will be released with patched firmware.) This is a tedious process by hand because 1) addresses and data are given in hexadecimal, 2) firmware is packed 2 MPG words per Multics word, using only 8 bits of each 9-bit byte, and 3) a new checksum must be computed.

Detailed Proposal
Install a program which can deal with firmware segments. Its two entries, patch_firmware, and dump_firmware, are described on the following pages.
The patch_firmware command is used to patch a segment containing an image of a firmware module for an MPC.

Usage

    patch_firmware path mem addr word1 ... wordn

where:

1. path is the path name of the segment containing the firmware.
2. mem is the memory overlay to patch. This argument may be "cs" to patch the control store overlay, or "rw" to patch the read/write memory overlay.
3. addr is the starting address to patch, in hexadecimal.
4. word1 are new MPC words, in hexadecimal. All wordi must be in the range 0-FFFF. Up to 16 words can be patched with one patch_firmware command.

Notes

The patch_firmware command will display the old and new contents of each firmware word patched, as well as the checksum, before the patch is made. The user is then asked whether the patch is correct. The patch will not be made unless the user answers yes.

Firmware modules may be retrieved from the firmware tape using the load_firmware_file command. Normally, firmware modules are kept in the archive >1dd>firmware>firmware.archive.

Firmware patches are usually given as absolute memory addresses in the MPC. Since the first 512 words (200 hex) are not supplied in the distributed firmware modules (they are hard-wired in the MPC), it may be necessary to relocate patches by subtracting 200 hex from the address given.
dump_firmware

Name: dump_firmware

The dump_firmware command is used to dump the contents of a segment containing MPC firmware.

Usage

dump_firmware path mem addr count

where:

1. path  
is the path name of the segment containing the firmware.

2. mem  
must be "cs" to dump the control store overlay, "rw" to dump the read/write overlay, or "size" to print the locations and lengths of overlays in the module. If this argument is size, no further arguments need be given.

3. addr  
is the starting address to dump, in hexadecimal.

4. count  
is the number of words to dump, in decimal.

Notes

Firmware modules may be retrieved from the firmware tape using the load_firmware_file command. Normally, firmware modules are kept in the archive >l|d|d|f|i|m|w|a|r|e|>firmware>firmware.archive.
SUMMARY:

Remove >unbundled from the search rules for the initializer process set up by sc_init_. Do not call com_err_ if an error occurs when initializing search rules.

REASONS:

Some sites do not have >unbundled. Therefore, the system distribution tapes do not contain the directory >unbundled. So when a new release is loaded, sc_init_ encounters an error setting up the ring 4 search rules for the initializer, and calls com_err_, which raises command_error_ and crashes the system.

IMPLICATIONS:

Simpler installation procedures.
### MULTICS CHANGE REQUEST

**MCR** 1979

**TITLE:** Fix bug in initiate_search_rules

**AUTHOR:** VanVleek

**Planned for System:** not applicable 5.0

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

**Documented In MTB:** not applicable

**Incompatible Changes:** no

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

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

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

**CATEGORY (check one):**
- Lib. Maint. Tools
- Sys. Anal. Tools
- Sys. Prog. Tools
- 355
- 905
- Salvager
- Ring Zero
- Ring One
- SysDaemon/Admin
- Runtime
- User Command/Subr

**DOCUMENTATION CHANGES (specify one or more):**
- MPM (vol,sect) MFAM (sect)
- MOSN ( sect) MSAM (sect)
- PLMs (AN#) an61
- Info Segs
- Other

**OBJECTIONS/COMMENTS:**

---

**SUMMARY:**

Fix a bug in initiate_search_rules which crashes the system if a directory which belongs in the user search rules is not found.

**REASONS:**

This error should result in failure to create the process, instead of a crash.

**IMPLICATIONS:**

If this error occurs operations will be able to create the directory from admin mode without losing the system.
**SUMMARY:**

Fix a bug in `salv_rebuild_directory` which leads to damage to links.

**REASONS:**

The salvager should not destroy links.

**IMPLICATIONS:**

Salvager will work.
## Title: Experimental IO Daemon Meters

### Author: J.C. Whitmore

- **Coded in**: [X] PL/I [ ] AIM [ ] Other...
- **Planned for System MR**: AK
- **Fixes Bug Number(s)**: NONE
- **Documented in MIB**: 355
- **User/Operations-visible Interface change?**: [X] Yes [ ] No
- **Incompatible change?**: [ ] Yes [X] No
- **Performance**: Better [X] Same [ ] Worse
- **Replaces MCR**: None

### Category (Check One)

<table>
<thead>
<tr>
<th>Document</th>
<th>Specify One or More</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td>MPM (Vol, Sect.) Commands</td>
</tr>
<tr>
<td>Sys. Anal. Tools</td>
<td>PLM (AN #) IO Daemon</td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td>MOSN (Sect.)</td>
</tr>
<tr>
<td></td>
<td>MPAM (Sect.)</td>
</tr>
<tr>
<td></td>
<td>IO Daemon</td>
</tr>
<tr>
<td></td>
<td>User Cmd/Subr.</td>
</tr>
<tr>
<td></td>
<td>Info Segs  IO Daemon changes</td>
</tr>
</tbody>
</table>

### Objections/Comments:

None (Reason)

### Use these headings:


**Summary:** Change `io_daemon` account to save some statistics on request processing. Data will be gathered per request type, per queue. Metering will be a site option. Data will include:
- number of pieces
- number of pages
- number of lines [blocks for cards]
- time of last request
- wait time of last request
- average wait time (exponentially averaged)

A follow up MCR will be submitted before installation to provide documentation of administrative and command interfaces.

**Reasons:** The temporary meters installed to compare the new and old charging algorithms proved to be very useful. MIT (IPC) requested the addition of the waiting time meters to help users choose the proper queue.

**Implications:** Aggregation of data at multilevel AIM sites will create a potential write down path.
<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>Document</th>
<th>Specify One or More</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sys. Anal. Tools</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Interface change?</th>
<th>yes</th>
<th>no</th>
</tr>
</thead>
<tbody>
<tr>
<td>Incompatible change?</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Performance:</td>
<td>Better</td>
<td>Same</td>
</tr>
<tr>
<td>Worse</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TITLE:** IO Daemon Emergency Fix

**AUTHOR:** J. C. Whitmore

**Coded in:** PL/I

**Planned for System MR:**

**Fixes Bug Number(s):** 355

**User/Operations-visible**

**Interface change?**  yes  no

**Incompatible change?** yes  no

**Performance:** Better  Same

**Replaces MCR**

**Expires:** 12/29/76

**DATE:** 6/21/76

**STTUS:** Written

**DOCUMENTATION CHANGES**

<table>
<thead>
<tr>
<th>Lib. Maint. Tools</th>
<th>MPM (Vol, Sect.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sys. Anal. Tools</td>
<td>PLMS (AN #)</td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td>MOSN (Sect.)</td>
</tr>
<tr>
<td></td>
<td>MPAM (Sect.)</td>
</tr>
</tbody>
</table>

**Objections/Comments:**

FCO #PHAPFG096 Fixes the hardware bug.

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

**Summary:** Add one line of code to output request which will make a dummy reference to the translate table. This will ensure that the table is in core when the real translations are done.

**Reasons:** A hardware bug causes the translate builtin to fail when the translate table straddles a page boundary and a page fault occurs during translation. This has caused the pathname normally printed on the head sheet to be blank.
**TITLE:** Fix bug in tape_ansi_

**AUTHOR:** M. Asherman

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>Status</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td>A</td>
<td>06/29/76</td>
</tr>
<tr>
<td>Sys. Anal. Tools</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**DOCUMENTATION CHANGES**

<table>
<thead>
<tr>
<th>Document</th>
<th>Specify One or More</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPM (Vol, Sect.)</td>
<td></td>
</tr>
<tr>
<td>PIIM (AN #)</td>
<td></td>
</tr>
<tr>
<td>MOSN (Sect.)</td>
<td></td>
</tr>
<tr>
<td>MPAM (Sect.)</td>
<td></td>
</tr>
<tr>
<td>MSAM (Sect.)</td>
<td></td>
</tr>
</tbody>
</table>

**SUMMARY:**

There is a bug in tape_ansi_ which prevents the control segment from being deleted on detach. The effect of this is to cause subsequent attachments to retain some of the old attach information, in spite of the new attach description.

**DETAILED PROPOSAL:**

Fix bug in tape_ansi_detach_.

**Objections/Comments:**

No objections/comments.

**Use these headings:** Summary of Proposal, Reasons for Proposal, Implications, Detailed Proposal.
# Multics Change Request

**Title:** Modify commands which truncate segments.

**Author:** VanVleck

**Planned for System:** not applicable

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

**Documented in MTD:** not applicable

**Incompatible Changes:** no

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

**Coded in:** ( ) PL/I ( ) ALH ( ) other—see below

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

## Documentation Changes (specify one or more)

<table>
<thead>
<tr>
<th>Change</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPM (vol, sect)</td>
<td>Lib. Maint. Tools</td>
</tr>
<tr>
<td>MPAM (sect)</td>
<td>Sys. Anal. Tools</td>
</tr>
<tr>
<td>MOSN (sect)</td>
<td>Sys. Prog. Tools</td>
</tr>
<tr>
<td>MSAM (sect)</td>
<td>355</td>
</tr>
<tr>
<td>PLMs (AN#)</td>
<td>BOS</td>
</tr>
<tr>
<td>Info Segs</td>
<td>Ring Zero</td>
</tr>
<tr>
<td>Other</td>
<td>Ring One</td>
</tr>
</tbody>
</table>

**None (reason): bug fix**

## OBJECTIONS/COMMENTS

---

**SUMMARY:**

Several commands and utility subroutines truncate segments to zero just before replacing their contents. Change these commands to truncate the segments to the new length after writing out the new contents.

**REASONS:**

If the new segment contents are the same length as the old, this change will save a VTOC write and a bunch of depositing and withdrawing of records. Avoiding these operations will make the system respond faster in real time and use fewer resources.

**IMPLICATIONS:**

System performance will improve.

System reliability is not affected by this change. But in a certain class of crashes, this change will cause a user's segment to contain a mixture of old pages and new pages instead of a mixture of zero pages and new pages.
DETAILED PROPOSAL:

The following programs will be modified:

archive, archive_sort, reorder_archive
qdx
indent
encode

sort_profile, sort_reqfile
clear_profile, clear_reqfile
rename_pnts_
save_dir_info
replacer
cpg_
**TITLE:** Correct debug interpretation of offsets in instructions with IC modification

**AUTHOR:** Susan Barr

**STATUS**

- **Written:** 06/22/76
- **Status:** A 06/29/76
- **Expires:** 12/29/76

**CATEGORY:** Lib. Maint. Tools

**MULTICS CHANGE REQUEST**

**TITLE:** Correct debug interpretation of offsets in instructions with IC modification

**AUTHOR:** Susan Barr

**STATUS**

- **Written:** 06/22/76
- **Status:** A 06/29/76
- **Expires:** 12/29/76

**DOCUMENTATION CHANGES**

**CATEGORY:** Lib. Maint. Tools

**MULTICS CHANGE REQUEST**

**TITLE:** Correct debug interpretation of offsets in instructions with IC modification

**AUTHOR:** Susan Barr

**STATUS**

- **Written:** 06/22/76
- **Status:** A 06/29/76
- **Expires:** 12/29/76

**DOCUMENTATION CHANGES**

**CATEGORY:** Lib. Maint. Tools

---

**SUMMARY:**

Correct print_text_ so that the offset of a word referred to by a break instruction that has IC modification will be computed correctly.

**PROPOSAL:**

Set a flag if the procedure was invoked via the break instruction entry. If this flag is set, use the real offset in computing the data offset.
**SUMMARY:**

Fix move, copy -acl and copy -name to print the reason for failure when names or acl cannot be copied due to no m access on the target's parent. Follow the error message that is currently printed:

"copy: Incorrect access ... target_dir>target_entry"

with the message:

"Unable to copy acl."

or

"Unable to copy names."
## TITLE: Make online_salvager_output reverse-deciduous

### AUTHOR: Bernard Greenberg

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: (W)PL/I ( )ALM ( )other-see below
Performance: ( )better (W)same ( )worse

### DOCUMENTATION CHANGES (specify one or more)

MPM (vol,sect) MPAM (sect) MDSN (sect) MSAM (sect) PLMs (AN#) Info Segs Other

### OBJECTIONS/COMMENTS:

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

### SUMMARY:
Causes the segment online_salvager_output to appear in the global hardcore address space. Store its bit count in its first word.

### REASONS:
The online salvager currently initiates >online_salvager_output in order to speak his mind. The online salvager is called only in times of distress; initiating segments in the root at these times is a poor policy, and has often resulted in mylock errors (root was already locked) and similar mishaps.

### IMPLICATIONS:
The copy_salvager_output (cso) command must be changed to develop the bit count of the online_salvager_output segment from its first word.

The online salvager also puts segments in >dumps: proposal to prevent mylocks in this instance will be supplied under separate cover.

### DETAILED PROPOSAL:
Reserve a hardcore segment number for online_salvager_output. Have init_branches (collection II initialization) locate >online_salvager_output, and activate it. Place an SDW for this useful segment in that SDW slot, thereby propagating it to all processes. Change salv_print_online to reference this segment by hardcore-prelinked symbol, as opposed to the storage system it is salvaging.