TO: Distribution
FROM: Joan Archer Scott
DATE: 29 September 75
RE: Multics Change Requests

Enclosed are copies of Multics Change Requests which were approved from 1 September 75 through 15 September 75.
# MULTICS CHANGE REQUEST

**TITLE:** Modify card reading software  

**AUTHOR:** Noel I. Morris

<table>
<thead>
<tr>
<th>Planned For System:</th>
<th>not applicable</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixes Bug Number(s):</td>
<td>not applicable</td>
<td></td>
</tr>
<tr>
<td>Documented in MTB:</td>
<td>not applicable</td>
<td></td>
</tr>
<tr>
<td>Incompatible Change:</td>
<td>yes</td>
<td></td>
</tr>
<tr>
<td>User/Operations-visible Interface Change:</td>
<td>no</td>
<td></td>
</tr>
<tr>
<td>Coded in:</td>
<td>(A)PL/I</td>
<td></td>
</tr>
<tr>
<td>Performance:</td>
<td>(N)better</td>
<td></td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>MPM (vol,sect)</th>
<th>MPAM (sect)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOSJ (sect)</td>
<td>MSAM (sect)</td>
</tr>
<tr>
<td>PL/Is (AN#)</td>
<td>AN 57</td>
</tr>
<tr>
<td>Info Jogs</td>
<td></td>
</tr>
<tr>
<td>Other</td>
<td></td>
</tr>
</tbody>
</table>

**OBJECTIONS/COMMENTS:**

**SUMMARY:**

The card reading software will be recoded in PL/I to be more efficient and to remove several bugs, glitches, and design errors. Elements sizes used by the different card reading DIMs will be changed to reflect a proper elements size for each application. These are summarized below.

**REASONS:**

The current card reading software is essentially coded in PL/I and is extremely inefficient. The element size used throughout is 1 bit, which is meaningless for such DIMs as raw and mcc. Some of the DIMs expect to be given an element count in characters, but they return a count of transmitted elements in bits! This is clearly confusing and should be illegal. The proposed recode will correct this problem.

**IMPLICATIONS:**

The card reading Daemon programs will have to be modified in order to correctly operate with the new element sizes supported by this new software. This is a minor change.

**DETAILED PROPOSAL:**

The mcc, viipunch, raw, and flipper DIMs will be rewritten. These DIMs will remain ios_modules. They will handle an element size
appropriate to their function. These are tabulated below:

<table>
<thead>
<tr>
<th>DIM</th>
<th>elsize</th>
</tr>
</thead>
<tbody>
<tr>
<td>mcc</td>
<td>9</td>
</tr>
<tr>
<td>raw</td>
<td>950</td>
</tr>
<tr>
<td>viipunch</td>
<td>1</td>
</tr>
<tr>
<td>flipper</td>
<td>9</td>
</tr>
</tbody>
</table>
MULTICS CHANGE REQUEST

TITLE: Extensions to syserr

AUTHOR: Larry Johnson

Planned for System: MR3.1
Fixes Bug Number(s): not applicable
Documented in MTB: 173
Incompatible Change: yes
User/Operations-visible Interface Change: no
Coded in: (X)PL/I (X)ALM ( )other-see below
Performance: ( )better (X)same ( )worse

DOCUMENTATION CHANGES

MPM (vol,sect) MPAM (sect)
MOSN (sect) MSAM (sect)
PLMs (AN#) AN65
Info Segs
Other

HEADINGS:

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

Summary

Provide interfaces to place binary data in the syserr log, and to expand error_table_ codes in the syserr message. These interfaces were originally described in MTB-173.

Reasons

1. Binary data (such as machine conditions or IOM status) is useful in reporting hardware problems. It is more efficiently handled if stored in binary instead of ASCII.

2. Several programs use syserr to print an error_table_ code in octal. These messages would be more meaningful if the text of the message were included.

Detailed proposal

1. Add the entry point syserr$binary called as follows:

   dcl binary_syserr_data(data_len) bit(36) aligned based(data_ptr);
call syserr$binary (action_code, data_ptr, data_len, ctl_string,
   arg1, ...., argn);

where data_ptr and data_len describe the binary data to be included in the syserr log message. This data would never be printed on the operator's console. Up to 128 words of data could be passed in this fashion.
2. Add the entry point syserr$error_code called as follows:

call syserr$error_code (action_code, error_table$code, ctl_string,
arg1, ......, argn);

The error_table_ code will be expanded as com_err_ does and appended
to the end of the regular syserr message. This entry must not be used
by procedures which are non-interruptable for page faults, since
error_table_ is not wired.

3. Change the format of the syserr log, both to accomodate the binary
data, and to allow for future syserr extensions.

4. Add a version number to the syserr log header. Current syserr logs
will be version 0; future logs will be version 1.

5. Modify print_syserr_log and daily_syserr_process to know about both
syserr log formats. A new control argument, -expand, will be added to
specify whether binary data is to be printed in octal or to be ignored.

6. The size of the wired syserr log will be increased from 150 to 400
words to allow for messages containing binary data. This will provide
enough space for two full length messages (256 characters + 128 binary
words).

7. Investigate lockless queueing techniques as used by ioi_ for handling
messages in the wired syserr log.

Implications

1. Programs which look at syserr logs will at least have to be recompiled
to work correctly.

2. The syserr log must be reset before booting a system containing these
features.
TITLE: Install any_to_any_ and dec_ops_ Bug Fixes

AUTHOR: R. A. Barnes

SUMMARY: Fixes 4 PL/I bugs.

DETAILED PROPOSAL:

Fix the following bugs:

1118 in which bit to char and char to bit conversions fail for strings longer than 256.

1297 in which flt_dec3 = "8.23e-127" gets conversion condition rather than underflow.

1301 in which ".00000000e0b" gets underflow while being converted.

1398 in which some cases of the mod, trunc, ceil, and floor builtins fail with decimal arguments. This bug was caused by a misunderstanding of the hardware specifications for the truncation indicator.
**TITLE:** Remove Default Initialization of Varying Strings from pll

**AUTHOR:** R. Barnes

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

**Objections/Comments:**

<table>
<thead>
<tr>
<th>Info Segs</th>
<th>None (Reason)</th>
</tr>
</thead>
</table>

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

**SUMMARY:**

Change the PL/I compiler to only perform varying string initialization that has been explicitly specified by the user.

**REASON:**

The generated code will be better and the compiler will run minutely faster.

**IMPLICATIONS:**

Although this is an incompatible change, the nonstandard initialization was never documented, so most user programs should not depend upon it. Some programs may stop working when recompiled.

**DETAILED PROPOSAL:**

The PL/I compiler has given all varying strings default initialization to "" or ""b at allocation time. Remove this feature.
A non-standard feature of the PL/I compiler has been that a varying string is initialized to "" or ""b upon allocation, regardless of whether or not the programmer has specified any initialization for the string. This feature will be removed shortly. Programmers wishing to continue having their varying strings initialized upon allocation as above should add an initial attribute to the declaration of the string.

(END)
TITLE: Add entry mailbox_update_message_index

AUTHOR: S. Herbst

CATEGORY (CHECK ONE)
- Lib. Maint. Tools
- Sys. Anal. Tools
- Sys. Prog. Tools
- Other

STATUS
- A
- B
- C
- D
- E
- F
- G
- H
- I
- J
- K
- L
- M
- N
- O
- P
- Q
- R
- S
- T
- U
- V
- W
- X
- Y
- Z

DATE
- 08/27/75
- 09/09/75
- 03/09/76

DOCUMENTATION CHANGES
- Document
- Specify One or More

EXPIRES
- 03/09/76

Objections/Comments:

Use these headings:

SUMMARY:
Add update_message_index entry point to mailbox_gate.
This entry point, analogous to message_segment_update_message_index, rewrites an existing message with another of equal length.

REASONS:
To make the message updating feature available. It is needed by the new mail and message sending commands.

IMPLICATIONS:
The code in ring 1 to perform message updating already exists but must be tested.
**TITLE:** Fix bug in copy_names  
**AUTHOR:** S. Herbst

<table>
<thead>
<tr>
<th>Category (Check One)</th>
<th>Status</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lib. Maint. Tools</td>
<td>Written</td>
<td>08/27/75</td>
</tr>
<tr>
<td>Sys. Anal. Tools</td>
<td>Expires</td>
<td>03/09/75</td>
</tr>
<tr>
<td>Sys. Prog. Tools</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Document:** Specify One or More

**DOC:** BOS  
**OBS:** MPM (Vol, Sect.)  
**SysDaemon/Admin.:** MOSN (Sect.)  
**Ring Zero:** PLMS (AN #)  
**Ring One:** MPAM (Sect.)  
**Runtime:** 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 copy_names and move_names to tell how they are used when erroneously called with no arguments.

**REASONS:**

Currently, they return without doing anything when called this way.
**TITLE:** Make create_dir recognize bad control args  

**AUTHOR:** S. Herbst

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

<table>
<thead>
<tr>
<th>Pending Changes</th>
<th>System</th>
<th>Specify One or More</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>MPM (Vol, Sect.)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PLMS (AN #)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MSGN (Sect.)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MPAM (Sect.)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MSAM (Sect.)</td>
</tr>
</tbody>
</table>

Objections/Comments:

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

**SUMMARY:**

Change the create_dir command to complain about a misspelled control argument rather than create a directory by that name.

**IMPLICATION:**

Incompatible change.

**PENDING CHANGES:**

The create_dir command has been changed to complain about a misspelled control argument instead of creating a directory by that name.
TITLE: Fix bug in set_lock

AUTHOR: VanVleck

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

DOCUMENTATION CHANGES (specify one or more)

MPM (vol,sect) MPAM (sect)
MOSN (sect) MSAM (sect)
PLI8s (AN#)
Info Segs
Other

None (reason) bug fix

OBJECTIONS/COMMENTS:

---

SUMMARY: Change set_lock_ to avoid calling timer_manager_ if the IPS "alarm" is masked.

REASONS: The set_lock_ routine is sometimes called when the alarm IPS is masked, for instance during attachment through mrd_. If the lock is found locked, set_lock_ will attempt to call timer_manager_$sleep, which will sleep forever. There is already code in set_lock_ to bypass calling timer_manager_ when running in ring 1; this change causes a check of the IPS mask and the bypassing of timer_manager_ if "alarm" is masked.

IMPLICATIONS: Daemon startup will be more reliable.
**Multics Change Request**

<table>
<thead>
<tr>
<th>TITLE:</th>
<th>Fix ips_masking bug</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTHOR:</td>
<td>R. Mullen</td>
</tr>
</tbody>
</table>

- Coded in [X] PL/I [ ] AIM [ ] other
- Planned for System MR
- Fixes Bug Number(s) 355
- Documented in MTB

**Category (Check One)**
- [X] Lib. Maint. Tools
- [ ] Sys. Anal. Tools
- [ ] Sys. Prog. Tools
- [ ] Documentation Changes

**STATUS**
- MCR 1369
- Written 09/02/75
- Expires 02/09/76
- Status A
- Document Specify One or More

**Objections/Comments:**
- None

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

**SUMMARY:**

ring_zero_peek to establish handler for "linkage_error" only, not "any_other".

**REASONS:**

If "alrm" or "cput" is signalled, that signal is lost so far as timer_manager is concerned. Typically an auto_ips_mask is set before these conditions are raised. Since ring_zero_peek does not reset the ips_mask, the process will never see another "quit", "alrm", or "cput". If the process ever calls sleep, it will never wake up. This happens about once or twice a month to the Initializer.

**IMPLICATIONS:**

None.
Summary:

Fix three problems in MCS:

1) Since private-line gl15's always appear to be "dialled up", perform one more experiment before signalling Multics that the device is ready.

2) Fix a problem in hsla_man with a certain type of stop_channel while the exhaust condition exists for a particular channel.

3) Put in a test for zeroing low core, when we should not.

Detailed Proposal:

coded in 355map.
TITLE: Update system exec_com's for MR3.0

AUTHOR: T. Casey

SUMMARY:

Update several system exec_coms to be compatible with new software to be released in MR3.0.

REASONS:

Several interface changes to commands used in these exec_com's have caused the latter to stop working.

IMPLICATIONS:

System administrators will have to merge MR3.0 changes with those made locally (if any), before 3.0 will run properly. Therefore, listings of the new versions, and compare_ascii's of those with the old versions, should be included or referenced in the SRB, to allow updates to be made before 3.0 tapes are loaded.

DETAILED PROPOSAL:

Changes are required to:

(acct_start_up admin biller master system_start_up).ec

Rename the current versions of those exec_com's in tools to:

MR2.2.<name>.ec

and install the new versions in tools, adding the names:

MR3.0.<name>.ec to them.