To: MTB Distribution

From: Karen MacKenzie

Date: 03/15/82

Subject: Splitting Up the Commands Manual

Comments on this MTB will be greatly appreciated. Please send them to:

The System M forum >udd>pubs>km>mtgs>coms

or

KMacKenzie.Pubs on System M
KMacKenzie.Pubs on MIT

or

Karen MacKenzie
Honeywell Information Systems
575 Technology Square
Cambridge, MA, 02139

or

Mail Stop MA22

or call:

617-492-9322
HVN 261-9322

INTRODUCTION

With Multics Release 9.0, the MPM Commands and Active Functions Manual (Order No. AG92) grew to a size of over 800 pages. It is, in a word, huge. The size of this manual is a subject of complaint from an increasing number of users. They find the manual cumbersome and unwieldy, and as a result, difficult to use. Of less importance, but still a consideration, is the fact that the manual is becoming more and more difficult for the documentation unit and the printer to handle. Pulling repro is a real chore, and production is increasingly time-consuming.

This MTB discusses a number of plans for addressing this problem, and offers a recommendation as to which plan would be best.

PLAN 1: SPLITTING BY FIRST LETTER

A simple way of addressing the problem would be to split up the commands into several smaller volumes according to the letters of the alphabet they begin with. Four volumes would probably do it. The first volume would cover letters A through D (213 pages), the second would cover letters E through L (207 pages), the third, M through R (213 pages), and the fourth, S through Z (140 pages).

This solution has the advantage of retaining the encyclopedic aspect of the manual, while making its physical aspect more appealing. It has the disadvantage of forcing users to continue buying and dealing with the entire set of command descriptions.

PLAN 2: REMOVING DUPLICATES

Another simple way of addressing the problem would be to remove all of the commands which are also documented in other manuals. This would cut the size of the Commands Manual by 103 pages.

The following commands (totaling 37 pages) are also documented in the Mail System Users' Guide (Order No. CH23):

```
accept_messages (3 pgs.)
defer_messages (1 pg.)
delete_message (1 pg.)
have_mail (1 pg.)
immediate_messages (1 pg.)
last_message (1 pg.)
last_message_sender (1 pg.)
last_message_time (1 pg.)
print_mail (2 pgs.)
print_messages (2 pgs.)
read_mail (10 pgs.)
send_mail (8 pgs.)
send_message (2 pgs.)
send_message_acknowledge (1 pg.)
send_message_express (1 pg.)
send_message_silent (1 pg.)
```

The following commands (totaling 18 pages) are also documented in the Multics Resource Control Users' Guide (Order No. CT38):

```
acquire_resource (2 pgs.)
assign_resource (3 pgs.)
cancel_resource (1 pg.)
list_resource_types (1 pg.)
list_resources (3 pgs.)
release_resource (1 pg.)
reserve_resource (2 pgs.)
resource_status (2 pgs.)
set_resource (2 pgs.)
unassign_resource (1 pg.)
```

The following commands (totaling 12 pages) are also documented in the Multics COBOL Users' Guide (Order No. AS43):

```
cancel_cobol_program (1 pg.)
cobol (4 pgs.)
display_cobol_run_unit (1 pg.)
expand_cobol_source (3 pgs.)
run_cobol (2 pgs.)
stop_cobol_run (1 pg.)
```

The following commands (totaling 4 pages) are also documented in the Multics Carry Facility Manual (Order No. AN76):

cancel_carry_request (1 pg.)
enter_carry_request (2 pgs.)
list_carrry_requests (1 pg.)

The following commands (totaling 2 pages) are also documented in the Multics SORT/MERGE Manual (Order No. AW32):

merge (1 pg.)
sort (1 pg.)

The following commands (totaling 3 pages) are also documented in the Multics Fortran Users' Guide (Order No. CC70):

fortran (2 pgs.)
set_fortran_common (1 pg.)

The pl1 command (7 pgs.) is also documented in the Multics PL/I Reference Manual (Order No. AM83).

The apl command (2 pgs.) is also documented in the Multics APL Manual (Order No. AK95).

The basic command (1 pg.) is also documented in the Multics BASIC Manual (Order No. AM82).

The fast command (1 pg.) is also documented in the Multics FAST Subsystem Reference Guide (Order No. AU25).

The gcos command (1 pg.) is also documented in the GCOS Environment Simulator Manual (Order No. AN05).

The emacs command (6 pgs.) is also documented in the Emacs Text Editor Users' Guide (Order No. CH27).

The qedx command (7 pgs.) is also documented in the qedx Text Editor Users' Guide (Order No. CG40).

The edm command (2 pgs.) is also documented in the Multics FAST Subsystem Reference Guide AND the Introduction to Programming on Multics (Order No. AG90).

If this plan was implemented, it would make sense to remove a number of other commands which aren't currently documented elsewhere, but should be. The mail command (4 pgs.) would be moved to the Mail System Users' Guide; the cobol_abs command (2 pgs.) would be moved to the Multics COBOL Users' Guide; the fortran_abs and set_cc commands (3 pgs.) would be moved to the Multics Fortran Users' Guide; the display_entry_point_dcl, display_pl1io_error, format_pl1, indent, and pl1_abs commands (24 pgs.) would be moved to the Multics PL/I Reference Manual; the runoff and runoff_abs commands (24 pgs.) would be moved to the Multics WORDPRO Reference Guide (Order No. AZ98). Doing this would save an additional 57 pages, bringing the total number of pages cut to 160.

This solution has the advantage of leaving the Commands Manual largely as it is, while reducing the number of pages by about 20%. It has the same disadvantage as Plan 1.

PLAN 3: SPLITTING BY FUNCTION (AND REMOVING DUPLICATES)

A more involved plan for addressing the problem would be to remove the duplicated commands as discussed above AND split up the remaining commands into a number of smaller volumes according to the functions they serve. Possible volume titles and the commands they would include are listed below.

Multics Commands for Process and Address Space Control

add_search_paths
add_search_rules
cancel_abs_request
delete_search_paths
delete_search_rules
dial *
echo *
enter *
enter_abs_request
get_system_search_rules
hangup *
hello *
initiate
list_abs_requests
login *
logout *
modes *
move_abs_request

necho *
new_proc
no_save_on_disconnect
print_proc_auth
print_search_paths
print_search_rules
program_interrupt
release
save_on_disconnect
set_search_paths
set_search_rules
slave *
start
terminal_type *
terminate
where
where_search_paths
MAP *
029 and 963 *

Commands followed by a "*" are for gaining access to the system and should probably be in a separate section.

Multics Commands for Storage System Manipulation

add_name
adjust_bit_count
archive
archive_table
attach_lv
branches
cancel_retrieval_request
change_default_wdir
change_wdir
check_iacl
compare
compare_ascii
contents
copy
copy_acl
copy_dir
copy_iacl_dir
copy_iacl_seg
copy_names
create

create_dir
decode
default_wdir
delete
delete_acl
delete_dir
delete_iacl_dir
delete_iacl_seg
delete_name
detach_lv
directories
directory
dump_segment
encode
enter_retrieval_request
entries
entry
equal_name
files
get_quota
home_dir
hunt
link
links
list
list_accessible
list_acl
list_iacl_dir
list_iacl_seg
list_mdir
list_not_accessible
list_retrieval_requests
lv_attached
master_directories
merge_ascii
move
move_dir
move_names
move_quota
msfs
nondirectories
nonfiles
nonmaster_directories
nonmsfs
nonnull_links
nonsegments
nonzero_files

nonzero_msfs
nonzero_segments
null_links
path
print
print_default_wdir
print_wdir
process_dir
rename
segments
set_acl
set_bit_count
set_iacl_dir
set_iacl_seg
sort_seg
status
strip
strip_entry
suffix
switch_off
switch_on
truncate
unlink
walk_subtree
working_dir
zero_segments

Multics Active Functions and Environment Tailoring Commands

abbrev
after
and
answer
before
binary
bool
calc
calendar
ceil
change_error_mode
check_info_segs
collate
collate9
contents
convert_ec
copy_characters

date
date_time
date_time_after
date_time_before
date_time_equal
day
day_name
decat
decimal
default
divide
do
equal
exec_com
exists
floor
format_line
format_line_nnl
general_ready
get_pathname
greater
help
hexadecimal
high
high9
hour
how_many_users
if
index
index_set
length
less
list_help
list_ref_names
long_date
long_year
low
lower_case
ltrim
manage_volume_pool
max
memo
min
minus
minute
mod
month

month_name
nequal
ngreater
nless
not
nothing
octal
on
or
picture
plus
print_auth_names
print_motd
progress
query
quotient
ready
ready_on
ready_off
repeat_query
reprint_error
resolve_linkage_error
resource_usage
response
reverse
reverse_after
reverse_before
reverse_decat
reverse_index
reverse_search
reverse_verify
rtrim
search
select
set_time_zone
string
substr
system
time
times
translate
trunc
underline
unique
upper_case
user
validate_info_seg

value_defined
value_delete
value_get
value_list
value_path
value_set
value_set_path
verify
who
year

This volume could be merged with the Multics Commands for Process and Address Space Control volume.

Multics Commands for Programmers

bind
create_data_segment
cumulative_page_trace
date_compiled
debug
exponent_control
page_trace
probe
profile
run
set_severity_indicator
severity
stop_run
trace
trace_meters
trace_stack

Multics Input/Output Commands

attach_audit
cancel_daemon_request
canonicalize
close_file
convert_characters
copy_cards
copy_file
detach_audit
discard_output
display_audit_file

```
dprint
dpunch
dump_segment
file_output
format_line
format_line_nnl
get_mode
io_call
line_length
list_daemon_requests
move_daemon_request
overlay
print_attach_table
print_request_types
set_tty
tape_archive
vfile_adjust
vfile_status
```

This set of volumes would also have to include a master index showing where each command was located.

This solution has the advantage of allowing users to buy only those command descriptions which they actually need. It has the disadvantage of not providing an encyclopedic repository for every command.

PLAN 4: PHOTOREDUCTION

There is a possibility of further reducing the Commands Manual to 65%. This solution has the advantage of retaining the encyclopedic aspect of the manual while improving its physical aspect. It has the disadvantage of making the manual unpleasant, if not impossible, to actually read.

RECOMMENDATION

Many developers, when asked how they feel about splitting up the Commands Manual, say they can't imagine not having an encyclopedia of all the commands. The problem with this is that the Commands Manual as it stands now is NOT an encyclopedia. Many commands are not included -- those in the MPM Subsystem

Writers' Guide (Order No. AK92), the Multics System Tools Manual (Order No. AZ03), the Multics Operators' Handbook (Order No. AM81), the MAM System Administrators' Manual (Order No. AK50), etc. In fact, the Commands Manual is more like a dumping ground for all those commands for which we never wrote individual manuals. We started something when we created manuals like the SWG and Tools -- a breakdown of the commands and subroutines into functional groups. The problem is that we never finished the process.

For this reason, I recommend that we pursue Plan 3, with some extensions. First, we should think of the functional volumes as independent manuals, not as pieces of the Commands Manual. The names should be changed to indicate this. For example, the Multics Commands for Storage System Manipulation volume should be called instead the Multics Storage System Manual. Second, we should include the MPM Subroutines Manual (Order No. AG93) in this plan. The subroutines should be split up into the same functional groups as the commands, and the appropriate subroutines added to each proposed manual.

Users often tell us that our documentation puts too much emphasis on straight command descriptions. They want to know more about the reasons for using commands, and they don't want to be bothered with descriptions of 200 commands when they only use 20. So, as a third extension, we should add some text to each manual that explains the area of Multics (for example, the storage system) the manual is about, and gives users some idea of the advantages and disadvantages of using various commands and subroutines. This would provide users with some of the motivational information they are always looking for.

With these extensions, the master index I mentioned earlier would not be necessary. Its function would be taken care of by the Index to Multics Manuals (Order No. AN50).

To give customers the option of also having all of the commands in one place, we could expand the Multics Pocket Guide -- Commands and Active Functions (Order No. AW17) to include subroutines, and commands not currently documented there, and have it serve as a mini-encyclopedia.

Finally, there is the possibility of having our cake and eating it too. If the development staff really wanted it, we could maintain an informal encyclopedia just for them, done on the diablo remote printer. This would be a truly complete encyclopedia, including every command and subroutine in existence. Since this manual would obviously be very large, we would make only three or four copies of it, to be kept at the documentation stations.

Then, if customers also showed an interest in this all-inclusive manual, we could explore ways of providing them with it -- alphabetic volumes (Plan 1), online segments, or even fiche.