Roach

Multics Task Report
Programming Development Office Staff Note

MTR - 005
PDOSN - 144

TO: Distribution

FROM: Gary C. Dixon

DATE: October 1, 1973

SUBJECT: Multics Library Maintenance Tools Project Status

I. Significant Events

1) IPC memo TOS-Q, which describes the upcoming changes to the Multics condition mechanism, was completed, approved by Bob Freiburghouse and Melanie Weaver (the implementers), and sent to the publishers. It should be released to users sometime this week.

2) The first maps of the Hardcore and Online Libraries have been generated by the new library_map (lm) command. This command builds a tree of nodes containing the status information for each directory, segment, archive component, and MSF in a library, alphabetizes a list of selected nodes by primary name, and then outputs the status of these nodes into a file.

3) The reorganization of the bound components and unbound segments of the Standard Service Library into more logical bound segments is now complete. The goals of this reorganization were: (1) to move many of the unbound segments into bound segments to reduce the actual number of segments in the library; and (2) to move some bound components to a more logical bound segment. In carrying out the second goal, many components of bound_misc_commands_ (which also contains subroutines) were moved to other bound segments. With the exception of exec_coms, absin segments, meter segments, and other assorted data bases, there is now only one unbound segment in this library, the edm command. As part of the reorganization, three new bound segments were created: bound_conversion_rtns_, which contains conversion commands and subroutines (eg, convert_date_to_binary_, cv_float, decode_clock_value_); bound_info_rtns_, which contains information-gathering commands and subroutines (eg, help, where, who, list_ref_names, mail, check_info_segs, print_motd, resource_usage); and bound_command_env_, which contains commands and subroutines which establish, maintain, or alter the command environment (eg, error_table_compiler,

---

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

ready, change_error_mode, reprint_error, set_com_line, program_interrupt, command_query_, answer, progress).

4) The Hardcore portion of the extended Star Convention is scheduled for installation is MSS 20.13. The Online portion and the extended Equal Convention will be installed shortly after MSS 20.13 goes in.

II. Task Status

We are currently planning the work which the Library Maintenance Tools Project will perform during the next year or so. The work can be divided into five different categories:

A) Enhancing update_seg, the Online Library updater.
B) Implementing a library descriptor for the Multics System Libraries, and converting existing programs to use this descriptor.
C) Creating or improving several miscellaneous library maintenance programs.
D) Making all Multics System Library segments accessible to all users in such a way that access can be audited and billed.
E) Documenting the library maintenance tools and procedures.

It is our plan to create detailed proposals describing the work which must be done in each of these areas, to publish these proposals as MTB's, to submit the proposal's for approval to the Multics Change Review Board, and to schedule their implementation. This planning effort will continue throughout the month of October, and will be our highest-priority task.

Much of the documentation work which was scheduled for September was interrupted by the planning effort described above. As mentioned in a previous task report, we have completed preliminary outlines for a Multics Library Maintenance Manual (LMM) and for an LMM Program Logic Supplement (LMM-PLS). Peter Kelley has now finished outlining the sections of the LMM-PLS dealing with update_seg and with its task compiler. All of these outlines will be published in an MTB, along with an implementation schedule. We plan to write these manuals over the next eighteen months, hopefully with some help from the Honeywell documentation group.

My work on library_map has progressed well, although it is about two weeks behind schedule. I have determined the basic structure of a library descriptor, a data base which describes the contents and organization of a library, and which defines methods for search, mapping, cleaning up, extracting segments from, and printing the segments of a library. As mentioned above, library_map created maps of the Online and Hardcore Libraries, using a preliminary version of the library descriptor

for the Multics System Libraries.

III. Work Planned

1) complete the planning of future work; write the proposals and MCRs for this work.

   A) update_seg. (Kelley, Dixon, Roach, Scherer)

   B) library descriptor programs. (Dixon, Roach, Scherer)

   C) miscellaneous tools. (Dixon, Roach, Scherer)

   D) user-accessible library. (Dixon, Roach, VanVleck)

   E) documenting tools/procedures. (Kelley, Dixon)

2) finish the library_map command, and its subroutines. (Dixon)

3) implement the library_print command, which will print the contents of selected library segments (eg, include segments, info segments, pt segments, bind listings, etc). (Dixon)

4) begin implementing some of the update_seg enhancements. (Kelley)

GROUP_____PDO: MULTICS SUPPORT GROUP_____ DATE_____ PAGE__1___/___2_____

PROJECT_____Library Maintenance Tools Project_____ AREA_____library descriptor tools_____

TASK DESCRIPTION	PERSONNEL	START	FINISH	M-W	CHANGES-STATUS
library_map (lm): code, document command which replaces msl_global format	Dixon	05/25/73 06/30/73	06/25/73 10/15/73	3	a version with few options works now
library_print: code, document command which prints groups of library segments (eg, include segs)	Dixon	03/19/73 10/02/73	04/01/73 10/30/73	1	
stop update_include from maintaining backup archives	Scherer	10/02/73 10/02/73	10/30/73 10/15/73	.2	Must be done in conjunction with library print
library_info: code, document command which replaces msl_info	Dixon			1.5	
library_get: rewrite gls to use library descriptor	Dixon			1	
cleanup: clean up the code; use library descriptor	Dixon			1	
library_object_cref (loc): rewrite cross_reference to use library descriptor				2	
library_include_cref (lic): rewrite icref to use library descriptor & source maps in standard obj segs				1.5	
upd_doc_task_: use library descriptor to format description of an installation	Kelley			.5	
lib_descriptor_: implement get_tree entry point (for library_map, library_info, library_print, library_get)	Dixon	02/28/73 04/07/73	06/15/73 10/15/73	.5	mostly done

GROUP_____PDO: MULTICS SUPPORT GROUP_____ DATE_____ PAGE__2___/___2_____

PROJECT_____Library Maintenance Tools Project_____ AREA_____library descriptor tools_____

TASK DESCRIPTION	PERSONNEL	START	FINISH	M-W	CHANGES-STATUS
lib_descriptor_: implement cleanup entry point (for library_cleanup)	Dixon			.3	
multics_system_libraries_: finish get_tree entry point (for library_info)	Dixon	10/25/73	11/15/73	.5	
multics_system_libraries_: code cleanup entry point (for library_cleanup)	Dixon			.3	