Roach

Multics Task Report MTR - 019

TO: Distribution

FROM: Gary C. Dixon

DATE: November 28, 1973

SUBJECT: Multics Library Maintenance Tools Task Report

I. Significant Events

1) An MTB entitled "Some Thoughts on File Handling" has been submitted for publication. This MTB discusses how Multics is deficient in its handling of single-segment and multi-segment files.

2) The program and runoff macros used to generate the task lists attached to this task report have been installed in the Multics Development Library. Anyone interested in generating such task lists should 'dprint >udd>m>lib>info>task_list_.macros.runout' or should type 'help -pn >udd>m>lib>info>task_list_'.

3) Bug fixes for compare_entry_names and msa_manager_ have been submitted for installation. An MCR proposing that the date_time active functions act as commands has been submitted to the MCR board.

4) This will be the last task report issued in the Multics Library Maintenance Tools Series. As work on the new Library tools comes to a self-imposed close, the Library maintenance tools development effort is being re-integrated into the Multics System Maintenance Group. Progress on tool development will be reported in Dave Jordan's task report.

II. Task Status

Peter Kelley and I have completed the design of two extensions to the Online Library Installation program, update_seg. The first allows large modifications (e.g., the PL/I compiler) to be updated into the Libraries as a single installation, rather than as several separate installations. The second increases the amount of information provided in the installation documentation generated by update_seg when a modification is installed. The list of segments and components being modified will be reformatted to save space and improve readibility. In addition, a prose description of each installation (including MCR numbers) will precede the reformatted descriptions of segment/component changes. This prose description will be inserted into the installation documentation by using a new -log control argument of the 'update_seg initiate' command. MCR's have been submitted for both of these extensions.

Pete has also upgraded the documentation for the update_seg command (including the new use of the -log option) prior publication of this documentation in the SPS.

Pete is continuing work on an MTB proposing several other changes to the update_seg command, including use of initial ACLs, setting of safety switches, implementation of named ACLs, updating MSFs and links, renaming library segments rather than having to replace them to change names, etc. This should be published during the next month.

In order to prevent ill effects on the Online Library maintenance tools from the forthcoming directory control changes, Pete Kelley is surveying the tools to identify and fix any code which requires the 'a' segment access mode. Calls to CACL primitives have already been removed from all Online Library tools, and the CACLs on the Online Library directories in >ldd have been removed. Dave Jordan's changes to the Hardcore Library updater eliminate calls to the CACL primitives from the Hardcore updater. When this new updater is installed, we will remove the CACLs from the remaining >ldd directories.

My work on the new library tools which use the library descriptor is proceeding steadily. The library descriptor is a directory containing a series of specially-named links to the directories of the Multics System Libraries (Online, Hardcore, Info, Include, MPM, BOS, 355, Salvager, etc), and a library descriptor program which controls the searching of these libraries. Taken together the library descriptor directory and search program define the contents, organization, and naming conventions used in the Multics Libraries. The exec_com which creates the descriptor directory is complete, and the search program is nearly so. It is our hope to eventually modify all of the library tools to use this descriptor so that changes to the library organization can be made with a minimum of modification to the tools.

I am currently working on three library descriptor tools: library_print: library_info; and library_map (previously called library_list). library_print writes the contents of info segments, bind segments, include segments, source archive components, etc, into a file which is suitable for printing. This command is almost complete. Roger Roach will be using it to generate the copies of info segments that he prints bi-weekly for distribution to various locations. The library_map command outputs a status list for selected Library entries into a file suitable for printing. The map entries are sorted by the primary name of each library entry, with cross-reference entries for secondary names and archive components. Besides the information returned by hcs_$status_long, the map entries can include object segment information (from object_info_), the system id for the system in which Hardcore and Salvager Library entries were last updated, and will shortly include ACLs and initial ACLs. library_map will replace the Hardcore and Salvager MSLs (Multics Segment Lists). The library_info command is an online interface for listing selected map entries. Its output is printed on the user's terminal, and is similar to that generated by library_map, although usually more brief in format and content.

III. Work Planned

1) continue the planning of future work: write the MTBs and MCRs for this work.

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

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

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

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

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

2) finish the library_info, library_map, and library_print commands, and their subroutines. (Dixon)

3) publish an SPS writeup for the update_seg command. (Kelley)

4) write MTB proposing further changes to update_seg. (Dixon, Jordan, Kelley)

5) write MTB outlining contents of the proposed Library Maintenance Manual. (Kelley)

6) write an MTB describing the SPS subroutine writeups and program logic sections which must be written to document the Online Installation system. (Kelley)

7) submit phase II of the extended Star Convention for installation. (Dixon)

8) find bug fix for hcs_$star_ and submit for installation. (Dixon)

9) rewrite runoff macros to improve their efficiency prior to submitting them to the Multics Development Library. Macros include the MCR generator, MPM header macros, and Length, Index, Reverse_Index, and Form_Line functions. (Dixon)

10) Incorporate comments on enter_daemon_request into a new MTB and publish for further comments. One change is renaming enter_daemon_request to enter_output_request. (Dixon)

11) write MTB proposing several extensions to convert_date_to_binary_. The extensions would be implemented by splitting convert_date_to_binary_ into a date string parser and two new user-accessible subroutines, encode_clock_value_ and decode_clock_value_. (Dixon)

IV. Unscheduled Tasks

Because this is the final Library Maintenance Tools Task Report, I am including a brief description of tasks which we would like to perform, but for which no manpower has been scheduled.

1) Library Descriptor Commands: rewrite existing commands to use the library descriptor; create new commands which use it.
   a) library_get, a replacement for get_library_segment.
   b) library_cleanup, a replacement for the cleanup command.
   c) library_include_cref, a replacement for icref.
   d) library_object_cref, a replacement for cross_reference.
   e) library_consistency, a new program which makes sure there is source for all object segments, etc.
   f) library_compare_maps, a new program which compares a saved copy of the status tree created by the library_map command with a new copy and outputs the differences.

2) update_seg enhancements, as described in the forthcoming MTB.

3) giving all users controlled, audited access to the source segments for the Online and Hardcore Libraries, and charging them for extractions from the Libraries.

4) Utility Programs
   a) compare_source, a program which parses an old and a new copy of an ALM or PL/I source program into two trees, and prints any differences in the parsed trees. Such a program would allow us to compare an un-indented old source with indented new source.

b) add_copyright improvements: allow more copyright formats (a MAC-only copyright, an IPC-only copyright, a Honeywell-only copyright, etc); handle more source languages (LISP, Fortran, etc); use the current date in new copyright notices, rather than a pre-determined date; when checking for the existence of a copyright notice in a program, accept any date in the notice, not just 1972;

c) validate_info_seg, a program which insures that info segments being installed are in proper format (e.g., begin with a date, end with (END), have lines of 72 or fewer characters, etc).

5) documentation of existing and new Library tools: update_seg and its subroutines; library descriptor tools; Library Maintenance Manual; Hardcore and Online Installation procedures; etc.

GROUP____________PDO: MULTICS SUPPORT GROUP__________ DATE___11/30/73_____ PAGE__1___/___2_____

PROJECT______Library Maintenance Tools Project______ AREA_________library descriptor tools_________

TASK DESCRIPTION	PERSONNEL	START	FINISH	M-W	CHANGES-STATUS	
library_info (li): code, document command which replaces msl_info (HIGH)	Dixon	10/05/73 10/05/73	10/31/73 12/15/73	1.5	Add options to command	*
library_map (lm): code, document command which replaces msl_global_format (HIGH)	Dixon	05/25/73 06/30/73	05/25/73 12/15/73	3	Add options to command	*
library_print (lpr): code, document command which prints groups of library segments (HIGH)	Dixon	03/10/73 10/02/73	04/01/73 12/07/73	1 2	prints segments now; make it print archive components, chase links, print MSFs, etc	*
library_brief_map (lbm): code, document replacement for Roach's segment_list (MEDIUM)	Dixon	12/07/73	12/30/73	.5		*
li/lm: get and output ACLs, initial ACLs, extra status (MEDIUM)	Dixon	12/15/73	12/30/73	1.5		*
li/lm/lpr: add sorting options to commands (LOW)	Dixon	01/01/74	01/15/74	.5		*
lm/lpr: gather and output statistics on Library segments (LOW)	Dixon	01/15/74	01/30/74	1.5		*
library_search_rules (lsr): code, document command which lists directories searched by li/lm/lpr for a given library (LOW)	Dixon	01/15/74	01/30/74	1.5		*
library_get (lgt): rewrite gls to use library descriptor		________	________	2		
library_cleanup (lcln): clean up the code: use library descriptor		________	________	1.5		
library_object_cref (loc): rewrite cross_reference to use library descriptor		________	________	2.5		

GROUP____________PDO: MULTICS SUPPORT GROUP__________ DATE___11/30/73_____ PAGE__2___/___2______

PROJECT_______Library Maintenance Tools Project______ AREA_________library descriptor tools_________

TASK DESCRIPTION	PERSONNEL	START	FINISH	M-W	CHANGES-STATUS
library_include_cref (llc): rewrite lcref to use library descriptor & source maps in standard obj segs				2	

GROUP____________PDO: MULTICS SUPPORT GROUP__________ DATE___11/30/73_____ PAGE__1___/___1______

PROJECT_______Library Maintenance Tools Project______ AREA____________Planning and Design___________

TASK DESCRIPTION	PERSONNEL	START	FINISH	M-W	CHANGES-STATUS	
update_seg: plan handling of large installations	Kelley Dixon		11/02/73		Done; MCR submitted	*
update_seg: Plan improvements in the automatic documentation of installations	Kelley Dixon Scherer Jordan	10/10/73 10/10/73	11/15/73 11/25/73		Done: MCR submitted	*
update_seg: write MTB describing planned changes to Online updater	Kelley	10/20/73 10/20/73	11/15/73 12/15/73		2/3 complete	*
user_accessible libraries: plan how to do this	Dixon Jordan Roach VanVleck				Awaiting manpower	
schedule documentation effort for Library maintenance tools	Dixon Jordan Kelley	01/01/74	01/15/74	1	MTB will be prepared	*