Subject: MCR-10034, library_pathname Enhancements

Author: Gary Dixon

Date: February 20, 2017

The library_pathname (lpn) command is a command/active function used often in maintaining Multics system libraries. It belongs to a set of tools detailed in the Multics Library Maintenance Program Logic Manual, AN80-00. lpn was actually written after this manual was printed; so it is documented primarily by an info segment: >doc>privileged>lpn.info. In spite of what the >doc>privileged pathname suggests, and the tool residing in >tools library, all these library maintenance commands except update_seg may be used by all users.

The library_pathname active function is particularly useful in writing exec_coms to examine library components, or rebuild bound segments in the library. However, the active function has one flaw not shared by the command interface. If several files match the selection specifications given in the command, all their pathnames are printed; however, the active function returns only the first matching pathname, without indicating that others matches exist. For example:

```
lpn -library source -library object bound_library_tools_.**.archive
>ldd>tools>source>bound_library_tools_.s.archive
>ldd>tools>object>bound_library_tools_.archive

r 20:37 1.336 15

string [lpn -library source -library object bound_library_tools_.**.archive]
>ldd>tools>source>bound_library_tools_.s.archive
r 20:38 0.959 15
```

Also, the companion library tools to library_pathname have control arguments to return information about the archive that contains a source or object component; or to return information about other components in the containing archive. For example:

```
library_info -lb source cv_ptr_.pl1 -container

1 bound_conversion_rtns_.s.archive                type:  archive
  cv_ptr_.pl1                                     path:  >ldd>sss>source
   contents modified:  11/24/86  1230.0
2 cv_ptr_.pl1                        component updated:  11/20/86  1404.4

r 20:44 1.141 14

library_info -lb source cv_ptr_.pl1 -components -pathname

1 convert_new_oncode_.pl1  path:  >ldd>sss>source>bound_conversion_rtns_.s.archive
1 cv_entry_.pl1            path:  >ldd>sss>source>bound_conversion_rtns_.s.archive
1 cv_ptr_.pl1              path:  >ldd>sss>source>bound_conversion_rtns_.s.archive

r 20:44 0.387 13
```

It would be useful to have both -container and -component controls in library_pathname.

- Reference Multics Change Tickets: http://multics-trac.swenson.org/ticket/42
  http://multics-trac.swenson.org/ticket/43

Proposed Changes

Change >ldd>tools>s>bound_library_tools_.s::library_pathname.pl1, as follows:

- Eliminate -first_match as the default for active function invocations.
- Add -entry (-et), -container (-cont), and -components (-comp) used in the library tools, which function as demonstrated below. -entry is the default, if none of these are specified.
- Change the active function to call get_shortest_path_ to shorten output from the active function, so more paths can be returned. Issue an error if the selected pathnames will not fit in the active function return string.

Change >doc>privileged>library_pathname.info: document the new and changed control arguments.

Here is an example of library_pathname with the new control arguments:

```
lpn -lb source cv_ptr_.pl1
>ldd>sss>source>bound_conversion_rtns_.s.archive::cv_ptr_.pl1

r 20:58 0.419 14

lpn -lb s cv_ptr_.pl1 -container
>ldd>sss>source>bound_conversion_rtns_.s.archive

r 20:58 0.237 13

string ([lpn -lb s cv_ptr_.pl1 -components])
>ldd>sss>source>bound_conversion_rtns_.s.archive::convert_new_oncode_.pl1
>ldd>sss>source>bound_conversion_rtns_.s.archive::cv_entry_.pl1
>ldd>sss>source>bound_conversion_rtns_.s.archive::cv_ptr_.pl1
r 20:58 0.339 13
```

Documentation

Documentation for library_pathname is shown below.

```
02/20/2017  library_pathname, lpn

Syntax as a command:  lpn {search_names} {-control_args}

Syntax as an active function:  [lpn {search_names} {-control_args}]

Function:  returns the pathname of one or more entries in a library.
Archive pathnames are returned in the case of archive component entries
in the library.

This command uses library descriptor and library search procedures, as
described in "The Library Descriptor Commands" of the Multics Library
Maintenance (AN80) manual.
```

```
Arguments:
search_names
   are entrynames that identify the library entries whose pathnames are
   to be returned.  The star convention can be used to identify a group
   of entries with a single search name.  Up to 1000 search names can
   be given in the command.  If none are given, then any default search
   names specified for the library_info command in the library
   descriptor are used.

Control arguments:
-descriptor desc_name, -desc desc_name
   provides a pathname or reference name that identifies the library
   descriptor describing the libraries to be searched.  If no
   -descriptor control argument is supplied, then the default library
   descriptor is used.  The initial default library descriptor
   describes the Multics System Libraries.
-library library_name, -lb library_name
   identifies a library that is to be searched for entries matching the
   search names.  The star convention can be used to identify a group
   of libraries to be searched.  Up to 100 -library control arguments
   can be given in each command.  If none are given, then any default
   library names specified for the library_info command in the library
   descriptor are used.

-search_name search_name
   identifies a search name that begins with a minus (-) to distinguish
   the search name from a control argument.  There are no other
   differences between the search names described above and those given
   with the -search_name control argument.  One or more -search_name
   control arguments can be given in the command.
-entry, -et
   returns pathnames for only the library entries that match one of the
   search names.  This is the default.

-components, -comp
   returns pathnames for all the components related to a matching
   library entry, in addition to output for the matching entry. (See
   "Notes" below.)
-container, -cont
   for an archived entry, returns pathname for the library archive
   that contains the matching entry, rather than pathname of the
   matching entry. (See "Notes" below.)

-all_matches, -amch
   returns the names of all entries matching the specified search
   names. (Default)
-first_match, -fmch
   returns only the name of the first entry matching a specified search
   name.

Notes: The -container and -components control arguments are provided
to facilitate information gathering on all library entries related to
a given bound segment.  When only one component of a bound segment
archive is matched, -entry returns the archive component pathname for
only this matching library entry; -container returns pathname of the
containing archive; -components returns archive pathnames of all
components in that archive.
```

Discussion of this MCR

Mike Grady raised an issue about changing the defaults of this command:

> Sorry, and I may be wrong, but isn't changing the default behavior of a command/active function a violation of first principles? Do we know that this change won't break any existing ECs that might use it?
>
> Mike

Mike was satisfied with the following response from the author:

> Your concern about changing an active function's default behavior is valid. However:
>
> - I have not found any exec_coms that use the library_pathname or lpn active functions. I searched >sss, >tools, >unb, >obs.
> - library_pathname is a tool, never documented to most customers except via an info segment. So most users have never heard of this command/af.
>
> I therefore think this change to default behavior is acceptable.
>
> In support of the change, I find it very disconcerting that the active function silently returns fewer items than an equivalent command, simply because command and AF defaults differ. This difference in defaults, and difference in output is unexpected.
>
> The original reason for this difference was that the active function return string was much smaller in 1981, when this command was written. The general opinion then was that the active function could not return all matching paths. It should therefore return only the first match. In 1983, the -first_match and -all_matches control argument were added, with -first_match remaining the default; but at least the user had an option to return all paths. But later on, the af return string was greatly lengthened.
>
> In a recent test, I asked lpn to return archive component pathnames for all the components in all four bound_pl1_.(1 2 3 4).s.archive and their associated object archives. All 389 fit in the current return string. That's sufficient to handle any reasonable use of the lpn active function. (Also, the AF reports an error if all paths won't fit in the return string.)
>
> In short, this seems like a reasonable exception to the "first principles". The original reason for the AF defaults differing from the command defaults is no longer valid; no adverse consequences are expected from the change in defaults.

Version History

Date	Revision	Author	Comment
2017-02-20	1.0	Gary Dixon	Initial draft of this MCR.
2017-02-26	1.1	Gary Dixon, Mike Grady	Added Discussion section to capture review comments and responses.