Subject: MCR10087, Addition of >doc>obsolete (obs) Library

Author: Eric Swenson

Date: 2021-02-18

Multics Ticket: http://multics-trac.swenson.org/ticket/218

MCR10085, proposed by Gary Dixon, calls for moving the `validate_info_seg` (`vis`) tool from its current location in `>tools`, to `>obsolete` (`obs`). This tool is replaced by the new `verify_info` (`vi`) tool. It also proposes renaming the `help` command and `help_` subroutine to `old_help` and `old_help_`, respectively, and placing these in the `>obs` library.

Because `verify_info` is a new tool, there may be times when a user may wish to compare the output and behavior of `verify_info` with the old output and behavior of `validate_info_seg`. Because the syntax of `validate_info_seg` is complicated and there are many control arguments, the info segment describing that tool should be retained. There is currently no obvious place to store info segments associated with executables in `>obs`, and therefore, MCR10085 proposes to add this info segment to a new documentation library, `>doc>obsolete` (`obs`).

This MCR calls for creating the new library directory (`>doc>obsolete`) with an additional name of "`obs`". Every change to the Multics libraries calls for an update of the `multics_libraries_` library descriptor (source for this is `multics_libraries_.ld`).

The Multics library descriptor `multics_libraries_` defines all the library directories and categorizes them under "library names" so that the various commands and active functions that leverage the library descriptor can choose the scope of the request.

For details of the Multics library descriptors, see the "Multics Library Maintenance Program Logic Manual" (AN80-00), as well as the info segments for the library commands: `library_cleanup`, `library_descriptor`, `library_descriptor_compiler`, `library_fetch`, `library_map`, `library_pathname`, and `library_print`.

During testing of the library commands, it became obvious that the current handling the commands for obsolete libraries was non-optimal. If you use the `lpn` command to locate, say, `tape_mult_read_.pl1`, for which there are two versions — one in the hardcore library and one in the obsolete library, the command returns both copies. That is arguably the correct behavior. And the active function does too — also correctly. But if you mistakenly embed an active function usage in a command line like this:

```
cpa [lpn tape_mult_read_.pl1] >sysbuild>mcrs>MCR10071>==
```

the resulting command line (after expansion) is:

```
cpa  >ldd>hard>source>bound_tape_mult_.s.archive::tape_mult_read_.pl1
>ldd>obs>source>bound_tape_mult_.s.archive::tape_mult_read_.pl1
tape_mult_read_.pl1
```

If you didn't know that there were two sources for this program in two different libraries, you would have been surprised to see that the hardcore version of the source and the obsolete version were compared by `cpa` and your provided second pathname to `cpa` was ignored.

What is really needed is a way to exclude the obsolete library segments, by default, but to include them when you really want to reference them with the library commands.

The library commands already provide for a `-library` control argument, so it would be possible to specify `obsolete_library` as the library when the intent was to reference a segment from the obsolete libraries. The default for the `-library` control argument, however, is `online_libraries`, which currently includes the obsolete libraries. So the only way to exclude the online library segments, is to explicitly specify all the other libraries. This would be tedious.

What is needed is for segments in the obsolete libraries not be included by default.

Proposed Changes

The first change is to create the new directory, >doc>obsolete with the additional name "obs". It will be initially empty, but MCR10085 will cause it to gain one or more segments.

The second change is to update multics_libraries_.ld to add a entry for the >doc>obs directory.

A third change is to change the various obsolete library directories to not be included in `online_libraries`.

A fourth change is to add the name `obs` to `obsolete_library`, for ease of typing and in keeping with the other short names.

Finally, a fifth change is warranted. Because `obsolete_library` is no longer part of `online_libraries`, it should be added, explicitly, to the default library for the `library_cleanup` command.

A `cpa` of the current and proposed multics_libraries_.ld follows, with the history comments and some formatting changes removed for clarity.

```
A60                 library name:       online_libraries offline_libraries;
Changed by B to:
B75                 library name:       online_libraries offline_libraries obsolete_library;


A180
A181      Root:                         (obsolete_library obs online_libraries online on "").(
source s "");
A182            path:                   >ldd>obs>source;
A183            search procedure:       multics_library_search_$source_dirs;
A184
A185      Root:                         (obsolete_library obs online_libraries online on "").
(object o "");
A186            path:                   >ldd>obs>object;
A187            search procedure:       multics_library_search_$object_dirs;
A188
A189      Root:                         (obsolete_library obs online_libraries online on "").
```

```
(execution x "");
A190           path:                    >obs;
A191           search procedure:        multics_library_search_$execution_dirs;
A192
Deleted by B, preceding:
B195      \014


A202                                     obsolete_library obs
Changed by B to:
B204                                     obsolete_library obsolete obs
Inserted in B:
B379      \014
B380      /*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
*  *  *  *  *  *  *  */
B381
B382      Root:                          (obsolete_library obsolete obs).(source s "");
B383           path:                     >ldd>obs>source;
B384           search procedure:         multics_library_search_$source_dirs;
B385
B386      Root:                          (obsolete_library obsolete obs).(object o "");
B387           path:                     >ldd>obs>object;
B388           search procedure:         multics_library_search_$object_dirs;
B389
B390      Root:                          (obsolete_library obsolete obs).(execution x "");
B391           path:                     >obs;
B392           search procedure:         multics_library_search_$execution_dirs;
B393
B394      Root:                          (obsolete_library obsolete obs).(info "");
B395           path:                     >doc>obs;
B396           search procedure:         multics_library_search_$list_info_dirs;
B397
B398      /*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
*  *  *  *  *  *  *  */
Preceding:
A377
A378
A379       End:  multics_libraries_;Note that the definitions of obsolete_library and obs already
exist.
```

The mbuild script follows:

```
                  MCR10087.mb         03/17/21  1600.6 pdt Wed



Description:

 A) Add Root for >doc>obs directory.

 B) Remove all obsolete_library roots from: online_library online and ""
    library groups.  obsolete_library roots must be selected by
    explicitly giving (obsolete_library obsolete obs) LIBRARY name; or
    (obsolete_library obsolete obs).DIRECTORY where DIRECTORY can be:
      source object execution info

 C) Move obsolete_library entries to very end of the Root list.  When
    selecting from multiple libraries (e.g., -lb *), items in
```

```
    obsolete_library are seldom desired, and should be listed as final
    selection.
 D) Add default -library obsolete_library  for library_cleanup command.
 E) Add short name obs for obsolete_library.


Installation_directory:  >sysbuild>mcrs>MCR10087;


 Build_script:             MCR10087.mb;


 Unbound_obj:              multics_libraries_               IN: tools  REPLACE;
     source:               multics_libraries_.ld             REPLACE            compiler: ldc;
```

Testing

Testing will involve running the various library commands on segments (temporarily) added to the >doc>obs directory and ensuring expected results. In addition, an mbuild script will be created that involves adding a dummy info segment to >doc>obs to make sure that everything works correctly.

Finally, some tests with the library commands and the tape_mult_read_.pl1 source, which exists in both hardcore and obsolete libraries, will be performed to make sure that the correct behavior is observed.

Documentation

The list of libraries doesn't appear to be documented anywhere — other than by invoking commands that manipulate the library descriptor. Consequently, no documentation changes are required other than the requisite entries in the Multics Software Release Bulletin that will accompany the next release of Multics.

The info segment multics_libraries_,info documents the multics_libraries_ library descriptor, but doesn't need any changes as a result of this MCR.

Version History

Date	Revision	Author	Comment
2021-02-18	1.0	Eric Swenson	Initial
2021-03-17	1.1	Eric Swenson	Updated for removing obsolete_library from online_libraries.