

Subject: MCR-10013, Add an input_history_ I/O module to bound_video_
Author: Gary Dixon
Date: May 20, 2016

Add a new input_history_ I/O module, and a command to attach/detach this I/O module, to the existing bound_video_ object segment. MTB-1000 describes the changes in detail. They may be summarized as follows:

- **input_history.pl1**: a new command to oversee attachment of the input_history_ I/O module to the user_i/o switch, while moving its usual window_io_ attachment to a uniquely-named (or user-specified) switch to be monitored by input_history_. This command validates input_history_ attach arguments, then moves the user_i/o attachment to a uniquely-named switch, and attaches the input_history_ I/O module to user_i/o.
- **input_history_.pl1**: a new stream_input_output I/O module that captures input lines as they are read via iox_\$get_chars calls; and adds 5 external edit requests to the window_io_ line editor for selecting prior input lines for re-input.
- **input_history_data.incl.pl1**: a new include file that defines the input_history_ attach data, and history segment lock file structures. It is used primarily by input_history_.pl1; and used superficially by the other two routines above.
- **window_line_editor.incl.pl1**: the existing include file that defines the line_editor_info structure passed by the window_io_ line editor to external editing requests. A new version 3 of this structure adds elements needed to enhance the window_io_ line editor interface to external editor requests.
- **window_io_iox_.pl1**: the existing code that implements the window_io_ line editor. Small enhancements to the line editor interface will be coded here. Also, the \$get_top_kill_ring_string_ entrypoint is added to remedy the problem described below for window_editor_utils_.alm.
- **ihctest.pl1**: a new command to test the external line editor requests in input_history_, without actually attaching the input_history_ I/O module. It simulates calls from the window_io_ line editor to the external editing functions in input_history_. This permits probe-level debugging of the input_history_ code while not interfering with normal input through the user_i/o switch.

- **window_editor_utils_alm**: the existing transfer vector that provides some window_io_line editor support routines to external editor requests. While testing the above approaches, a major problem was found in the design of the window_editor_utils_\$get_top_kill_ring_element. There is no way to tell the actual length of the returned string, or whether the char(*) argument supplied to hold the kill ring string was long enough to hold the entire string. A new \$get_top_kill_ring_string will be added as a remedy.
- **bound_video_bind**: the existing bind file changed to add the new input_history_-related objects to the bound_video_ object.
- **window_editor_utils_info**: changes to describe the new \$get_top_kill_ring_string entrypoint.
- **input_history_info**: describing the new input_history_ I/O module.
- **input_history.info (ih.info)**: describing the user interface for the new input_history command.
- **video_editing.gi.info**: changes adding information about the input_history_edit request key bindings, when input_history_ is in use.