To: Distribution

From: James R. Davis

Date: 01/23/81

Subject: Menu Presentation Commands

1. INTRODUCTION

This MTB proposes commands and active functions for menu presentation based on the menu subroutines proposed in MTB 493. These commands are appropriate for light to medium use of menus. If efficiency is a concern, programmers should use the PL/I subroutines, or write other commands.

To send comments

by continuum, on System M
>udd>m>jrd>mtgs>menus

by Extended Mail Facility, either site
JRDavis.Multics

by HVN
261-9382

by Bell Telephone
(617) 492-9382

By USMail
c/o CISL
575 Tech. Sq. #3
Cambridge MA 02139

2. OVERVIEW

I propose two commands and one active function here. The first command, create_menu, creates a menu and stores it in a segment. A menu is displayed using with the display_menu command. A menu choice is obtained by the get_menu_choice active function (also a command).

These commands use the menu_ subroutines, and in turn the video system.

The caller sets up the video system.

These commands do not access all functions provided by the subroutines, since that would make the command interface too complex. Rather, they make it possible to use menus in fairly straightforward ways. The subroutines are available for those desiring other interfaces.

Documentation follows:

SYNTAX AS A COMMAND:

create_menu menu_path {control_arguments}

FUNCTION: creates a menu description, and stores the description in a segment. The menu may be used with the other menu commands, active functions, and subroutines.

ARGUMENTS:

menu_path
is the pathname of the segment to store the menu in. The segment is created if it does not exist. The old contents are destroyed.

CONTROL ARGUMENTS:

The first three control arguments specify the content of a menu. They may be given many times, since menus have many choices, and often several headers and trailers. In each case, a string argument is required, which must be quoted if it contains spaces or other special characters.

-option STR, -op STR
specifies a menu option. The options appear in the menu in the order given. At least one option must be supplied.

-header LINE, -he LINE
specifies a line of header. All header lines specified appear in the menu in the order given.

-trailer LINE, -tr LINE
specifies a trailer line. All trailers appear in the menu in the order given.

The remaining control arguments control the format of the menu. All are optional.

-columns N
Where N is a positive decimal integer, sets the number of columns in the menu to N. The default is one column.

-center_headers
causes all header lines to be centered.

-no_center_headers
causes header lines to be flush left. This is the default.

-center_trailers
causes all trailers to be centered.

-no_center_trailers
causes trailers to be flush left. This is the default.

-pad C
where C is one character, specifies the padding character for centering. The default is the space character.

-line_length N, -ll N
where N is a positive decimal integer, specifies the line length for the menu. If not supplied, the line length will be the line length of the user's terminal at the time the command is invoked.

ACCESS REQUIRED:

"w" on the segment.

SYNTAX AS A COMMAND:

```
display_menu  MENU {WINDOW}
```

FUNCTION: displays a menu in a window.

ARGUMENTS:

MENU
is the pathname of a segment containing a menu.

WINDOW
is the name of an I/O switch for a window. The default is user_output.

get_menu_choice, gmc

SYNTAX AS A COMMAND:

get_menu_choice MENU {WINDOW}

SYNTAX AS AN ACTIVE FUNCTION

[get_menu_choice MENU {WINDOW}]

FUNCTION: Given the menu MENU on display in window WINDOW, gets a menu choice from the user and returns it.

ARGUMENTS:

MENU
is the pathname of a menu on display in the window.

WINDOW
is the I/O switch of a window. The default is user_output.

NOTES:

For an option choice, the choice returned is a character string representation of a decimal integer (e.g. "3"). For a function key, the string returned is an integer prefaced by the letter "F" (e.g. "F3").