V I S U A L 200 K E Y P A D

cursor control keys -- standard pointer movement

RETURN	=	M	Move to start of next line
HOME	=	L*	Move to start of current line
BS	=	G-	Recover deleted line
LF	=	1	repeat last typed command
TAB	=	N	Next word/unit
SHIFT+TAB	=	N-	next word/unit backwards

	7 IL	8 IC	9 ST
	G* Insert lines	I! Insert chars	I . . Insert space
CF		I-* Recover chars	0-* Undo
	4 DL	5 DC	6 CT
	K Delete line	E Erase char	D . . Delete space
CF	K- Delete last line	<not available>	D-. . Delete space backwards
	1 EL	2 EF	3 EP
	E* Erase to end of line	S! Substitute text	^ Set Marker
CF	E-* Erase back to start of line	S" Substitute same	$ Switch Inputs
	0	, CPY	. PRT
	F! Find text	F" Find same	%D re-write Display
CF	<not available>	?	| Toggle destruct.
	-	ENTER	
	M-* Move back to top of file	\ Switch between command and data-entry mode	

The entries on the lines marked CF require simultaneous use of the CONVERT FUNCTION key.

Function keys F0 - F2 (but not keys F3 - F13) are also eligible for definition on those terminals which have them.

COMMAND CHECKLIST

A	Adjust line length
B	Break line in two at pointer position
C	Case-change with right-shift
D/.../	Delete first occurrence of <text>
D-/.../	Delete prior occurrence of <text>
E	Erase character to right of pointer
E-	Erase character to left of pointer
F/.../	Find first/next occurrence of <text>
F-/.../	Find prior occurrence of <text>
G/.../	Get (insert) <text> as complete line above current line
I/.../	Insert <text> to left of pointer
J	Join next line to current
K	Kill (delete) current line
K-	Kill (delete) previous line
L	Left-shift one character position
<	Cursor Left
M	Move forward one line
}	Cursor down
M-	Move back one line
{	Cursor up
N	Next -- locate next word
N-	Next back -- locate previous word
O/.../	Overwrite existing text with <text>
P	Print line on terminal
Q	Query Form (check spelling)
R	Right-shift one character position
>	Cursor Right
S/.../	Substitute <text> for text last found
T/.../	Traverse first occurrence of <text>
U/.../	Uncover first/next occurrence of <text>
V/.../	Verify presence of <text> at pointer position

@n	Align to column <n>
+n	Increment number by <n>
#n	Move to absolute line number <n>
$	Switch Input from main to secondary input stream, or vice versa.
^	Set Marker
^1,...,^6	Define Macro letter 'X','Y','Z','x','y','z'
=	Revert to Marker
¦	Toggle Destructive mode

Special commands

%A	Abandon edit without updating file
%C	Close edit normally
%CI	Close and apply IMP syntax check
%CP	Close and apply Pascal syntax check (Not yet available).
%D	re-write Display
%Dn	set minimum Display (MINWIN) to <n> and re-write
%E	alter Environment options
%G file-name	Get commands from file
%H	obtain Help information
%K	Key definition
%Ln	set Line width (WIDTH) to <n>
%Mn	set left Maragin (MARGIN) to <n>
%P file-name	Put key definitions to file
%Q	Query key definitions
%S file-name	define Secondary input file
%Wn	Wipe out record of last <n> deleted lines

Reserved symbols

( )	command grouping parentheses
,	separator for alternatives
\	suffix to invert failure condition
?	suffix to cancel failure condition
"	'ditto' text indicator
!	(initial) prefix for system command
!	(non-initial) direct-entry text indicator
%	prefix for special commands

Edinburgh University Computer Science Department M E M O

V E C C E on E M A S

These notes refer to the video-oriented version of the context editor ECCE. At present, this version is supported by the Computer Science Department on the 2972 and may be accessed via the Departmental library. Users who have not already done so should issue the once-only command:

OPTION SEARCHDIR=ECSLIB.GENERAL

or OPTION(SEARCHLIB=ECSLIB.GENERAL) if anyone still uses parentheses.

This makes available (inter alia) the system-level commands "VECCE", "VSHOW" and "VRECAP".

In general the basic Editor commands are the same as in conventional ECCE, with the addition of 'O' for Overwrite, and RETURN as an immediate-action equivalent to Move. Other immediate-action keys, such as Cursor Control, Linefeed, and Function Keys, cannot currently be used over the network, pending changes in TCP software. Also, at present, it is desirable to avoid typing ahead since this can easily mess up the screen image -- use %D to re-paint if it happens.

The command %H may be used to obtain help information (through VIEW).

The command %E is available to select various environment options.

Terminals

The Editor makes use of the value selected by TERMINALTYPE to establish the kind of terminal in use. The strategy used to update the screen depends on the capability of the terminal, in particular whether it can achieve split-screen scrolling and whether it can highlight an individual screen position independently of the input video cursor. Not all of the videos mentioned below have been fully tested and other suitable terminals can be added if required. Feedback from users would be helpful.

Hard-copy devices and 'glass-Teletype' videos are handled in much the same way as in conventional ECCE.

Perkin-Elmer Bantam: Lacks split-screen scroll capability.
File pointer indicated by 'splodge' temporarily overwriting character to left of current position or caret temporarily overwriting character underneath current position (user-selectable option).

VT52: Lacks split-screen scroll capability.
File pointer treatment similar to Bantam.

Visual 200: Has split-screen scroll capability.
File pointer indicated by displaying character at current position at reduced intensity (vertical bar if current position is blank).

Hazeltine Esprit: Has split-screen scroll capability.
File pointer indicated by displaying character at current position at increased intensity (vertical bar if current position is blank). This pre-supposes that the High-Intensity Enable switch at the rear of the terminal is ON.

Newbury: Has split-screen scroll capability.
File pointer indicated by underlining current character position. This pre-supposes that the Attribute Underline has been selected in Setup mode.

VT100: Has split-screen scroll capability.
File pointer indicated by displaying character at current position in reverse video.

Volker Craig 404: Lacks split-screen scroll capability.
File pointer treatment similar to Bantam.

ICL KDS7362: Has split-screen scroll capability.
File pointer treatment similar to Visual 200.

Hamish Dewar November 1982

Department of Computer Science
Edinburgh University

Hamish Dewar
April 1983

Video Version of ECCE

A video-oriented version of the context editor ECCE is now available on EMAS 2900 and VAX/VMS. It offers most of the functions of a screen editor while retaining (and in some respects extending) the general editing capability of ECCE.

This version also includes facilities for checking the spelling of words in text files, and for checking the syntax of IMP and Pascal programs. The spelling check matches words against a limited word-list comprising about 15,000 entries, the Kent/Edinburgh lexicon. The command "Q*" (or "Q0") locates the first/next word in the file which is not in the word-list. Suggestions for additions to the lexicon should be addressed to Neil Hamilton-Smith, ERCC, though it should be noted that the list is confined to words with a reasonable frequency of occurrence in written text. At present, the spelling check is independent of the grammatical context.

Program syntax checking is invoked by variants of the Close command. This is a novel feature and feedback from users would be helpful in refining it. Either of the Special commands "%Ci" or "%CI" has the effect of causing a check on the syntax of an IMP program file before it is finally closed. The first form (lower-case 'i') signifies that a return is to be made to the editor whenever an error is detected; the second form (upper-case 'I') implies checking through the entire file without interaction. The dialect of IMP which the checker accepts is the agreed common subset of EMAS IMP80 and VAX/VMS IMP77. Some, but not all, of the features which are peculiar to one of these dialects are also accepted, in general with a "Non-standard" warning. The checker detects most of the faults which would be reported by the compilers, though there may be minor discrepancies due to organisational differences and some of the table limits are inevitably different.

Pascal syntax checking will be added at the end of April. The checker will accept ISO standard Pascal, together with a few UCSD and Vax Pascal extensions.

Calling the Editor on EMAS

On EMAS, the program and related video package are held in ECSLIB. Users who do not already have this library inserted should give the command OPTION(SEARCHDIR=ECSLIB.GENERAL). The command name is VECCE. By default, the Editor assumes the terminal in use to be as defined by TERMINALTYPE; this may be over-ridden by specifying an alternative number from the ERCC enumeration as the value for the parameter TTYPE when calling the Editor.

Calling the Editor on VAX/VMS

On the Computer Science Vax, the Editor is an installed utility available under the name VECCE or the abbreviation V. The Vax version assumes Visual 200 by default. For any other terminal, it is necessary to supply the appropriate number from the ERCC enumeration as the value for the qualifier TTYPE when calling the Editor. The control key definitions for the Visual 200 which are in force at the outset are

shown below; suggestions are invited for possible alternative sets which might be of greater general-purpose utility.

Mode of connection

The most important factor affecting editor performance is the mode of connection of the user's terminal. In general, the more direct the connection, the better the performance will be. When accessed from some of the remoter parts of the ERCC network, the behaviour will be distinctly less satisfactory.

In addition to affecting the responsiveness of the editor, an indirect connection over the network involves an increased possibility of unintended modification of the data stream sent to the terminal, leading to an erroneous screen image which does not match the state of the file. It may also rule out the use of cursor controls and function keys.

If the screen image looks curious at any time, the "%D" command should be used to re-display it.

Network Terminal Controllers

New software on some of the Terminal Controllers on the ERCC network provides a much improved screen handling mode which the editor uses if available. On other TCPs, as an interim measure, interference with data may be reduced by entering TCP Graph mode and input of control characters may be enabled by entering Z mode. With the old TCP software, the screen image is particularly susceptible to corruption as a result of typing ahead; this should be avoided as much as possible.

***** NB *****

With the new TCP software, while the editor is being run, the system Attention character is changed from Escape to Null, which is keyed by CONTROL+@. This is done to allow the use of Escape sequences from the terminal for editing control. If the Attention key is pressed at any time, the screen-handling mode of the TCP is immediately cancelled, so that it should be used only in extremis.

Supported Terminals

The list of terminals currently supported as videos in ECCE is as follows (ERCC enumeration):

code	terminal	highlight	part-screen scroll
6	Perkin-Elmer Bantam	-	-
8	DEC VT52	-	-
11	Visual 200	reduced	yes
12	VT100	reduced	yes
13	Hazeltine Esprit	increased	yes
15	Newbury 8000	-	yes
20	Volker Craig 404	-	-
21	ICL KDS7362	reduced	yes