



Getting Started With JOVE

Contents

1. INTRODUCTION	3
1.1. INVOKING JOVE.....	3
1.2. TYPING IN TEXT.....	4
2. JOVE COMMANDS	4
2.1. MOVING THE CURSOR AND SCROLLING	5
2.2. DELETING TEXT.....	5
2.3. NUMBERING LINES	6
2.4. MOVING AND COPYING TEXT.....	6
2.5. SAVING AND EXITING	7
2.6. LOADING FILE CONTENTS INTO BUFFERS.....	7
2.7. PARAGRAPH AND LINE COMMANDS	8
2.8. SEARCH AND REPLACE	8
2.9. MANIPULATING BLOCKS OF TEXT	8
2.10. CANCELLING A COMMAND	9
2.11. REPEATING AN ACTION	9
2.12. MULTIPLE BUFFERS	9
2.13. MULTIPLE WINDOWS.....	10
2.14. MACROS	10
2.15. GETTING HELP	10

1. Introduction

JOVE is a streamlined derivative of the EMACS editor written at MIT by Richard Stallman. JOVE retains EMACS' powerful editing features, without suffering its disadvantages, namely its size and its dependence upon LISP. Unlike the EMACS editor, JOVE does not strive to be a full programming environment, but is instead pared down to the functionality of an efficient editor. Jove's compactness renders it an appropriate companion for Elate^{RTM}. The name JOVE stands for "Jonathan's Own Version of EMACS." JOVE is compatible with EMACS, but it should not be assumed that the two editors will invariably behave in the same way.

JOVE allows the user to edit files in a full text-based window using a complete set of keyboard editing commands. It is considered a 'screen' editor since the arrow keys may be used to move around the screen, and since, when text is inserted or deleted, the whole screen adjusts to reflect the change. JOVE is a multi-window, multi-buffer editor which provides on-line help, and support for spell-checking and program debugging.

1.1 Invoking JOVE

To start the JOVE editor from the shell, the user should type in

```
jove
```

to create a new file, or

```
jove <filename>
```

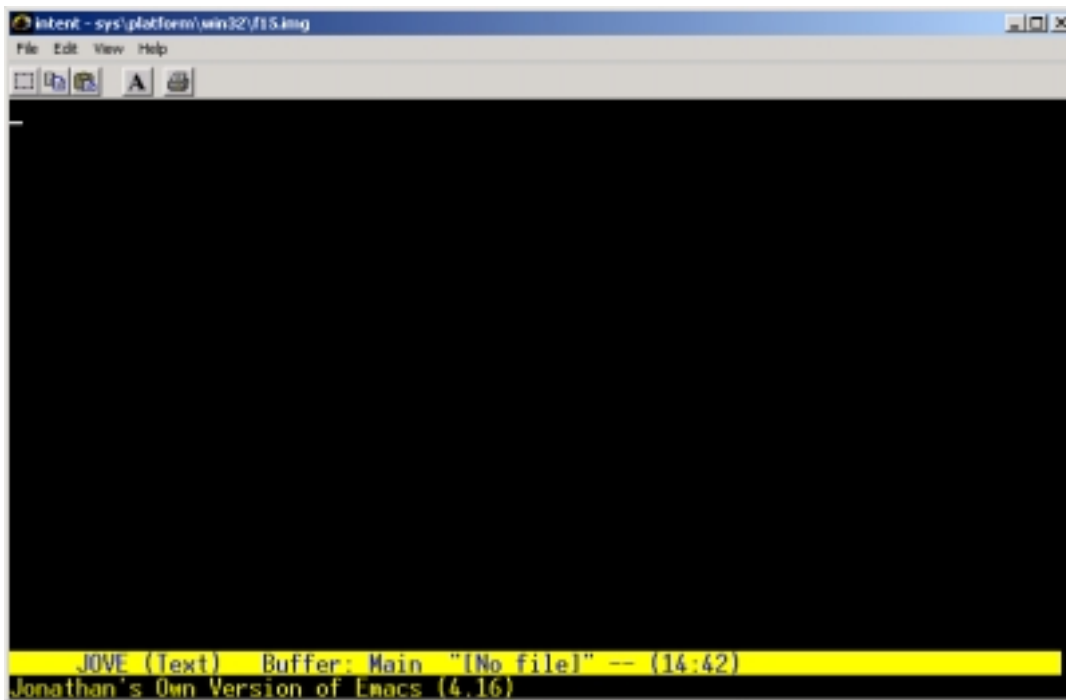
to edit an existing file of the specified name.

When using JOVE, all text is initially typed into a "buffer." This is a temporary copy of the document, which can be edited and ultimately saved as a permanent file. If the command "jove" is followed by no arguments, then the user will be given an empty buffer, called MAIN.

Any arguments that follow the command "jove" will be interpreted as file names, and the system will allocate a buffer to each. Only the first file is actually read into memory. The other files are only read in if the user attempts to use the buffers they have been allocated. This enhances JOVE's efficiency, since usually the editor will be run on a large list of files, only a few of which are actually edited.

A special JOVE buffer known as **minibuf** is used to store the names of all files specified on the command line. A user who is being prompted for a file name may choose to type Ctrl N and Ctrl P, and thus to cycle through the list of file names stored in this mini buffer. Under these circumstances, the file name is inserted at the place the user is typing, and may be edited as if it had been typed by the user.

Once the editor has been invoked, the JOVE screen will appear. Most of the screen will be taken up by a large blank area into which text can be entered. At the bottom of the screen lies the status line, or "mode line." This line gives information upon the name of the file being edited, the current text mode, the size of the file in characters and bytes, and whether the file has been modified.



In the example above, the mode line is that which reads:

```
JOVE (Text) Buffer: Main "[No file]" -- (09:18)
```

Below the status command lies a command line, where the user may enter various commands, and where the editor can output information in response to such commands.

1.2 Typing in Text

After the editor has been invoked, the user may immediately begin typing text into a buffer. If the cursor, or "point" is positioned in the middle of text already occupying the buffer, new text that is typed will be inserted at this location.

2. JOVE commands

JOVE makes use of an extensive set of keyboard commands, which may be modified by the user.

Some JOVE commands can be issued via special key strokes involving the Control (<Ctrl>) and Escape (<Esc>) keys. In the following list of commands, the notation <Ctrl-a> signifies "while holding down the Control key, press a." The notation "<Esc> a" means "press the Escape key briefly, then press a."

Other commands are entered at the command line at the bottom of the screen. The cursor may be moved to this location by typing:

```
<Esc> x
```

The command can then be completed by typing one or more command words and pressing <Enter>. All command functionalities may in fact be accessed through the use of commands typed at the command line, but the <Ctrl> and <Esc> keystrokes have been made available as shortcuts for the more commonly used commands.

Getting Started With JOVE

If a command is incorrectly typed, or the user attempts to issue an order that the editor does not recognise or will not allow, a beep will sound. JOVE will respond to other types of error by displaying an informative message at the bottom of the screen.

Besides the listed shortcuts, the functionality of several common commands may be accessed by pressing keys in the number pad (while <Num Lock> is disabled). The keys on the number pad are listed below, with the commands for which they supply shortcuts. The usual shortcut for each command is also given.

Number pad key	Command	Shortcut
1	end-of-line	<Ctrl e>
2	next-line	<Ctrl n>
3	next-page	<Ctrl v>
4	backward-character	<Ctrl b>
5	-	-
6	forward-character	<Ctrl f>
7	beginning-of-line	<Ctrl a>
8	previous-line	<Ctrl p>
9	previous-page	<Esc> v
0	over-write-text	none

2.1 Moving the Cursor and Scrolling

The arrow keys can be used to move the cursor within the text in the indicated direction. In addition, the following commands can be used to manipulate the cursor.

<Esc> f	Moves the cursor forward one word.
<Esc> b	Moves the cursor backward one word.
<Ctrl-a>	Moves the cursor to the beginning of the line.
<Ctrl-e>	Moves the cursor to the end of the line.
<Esc> a	Moves the cursor to the beginning of a sentence.
<Esc> e	Moves the cursor to the end of a sentence.
<Esc> ,	Moves the cursor to the first line on the current screen.
<Esc> .	Moves the cursor to the last line on the current screen.
<Esc> <	Moves the cursor to the first line of the entire buffer.
<Esc> >	Moves the cursor to the last line of the entire buffer.
<Ctrl-z>	Scrolls forward one line.
<Esc> z	Scrolls backward one line.
<Ctrl-v>	Scrolls forward one screenful.
<Esc> v	Scrolls backward one screenful.
<Ctrl-l> (the letter l)	Scrolls the screen so that the current line is centred vertically within the window. If the line is already in place, the screen is cleared and redrawn. This command may be preceded by the <Escape> key and a numeric argument, which specifies the offset from the top of the screen at which the line is to be positioned. <Esc> 0 <Ctrl l> will position a line at the top of the screen, <Esc> 1 <Ctrl l> will position it one line down from the top, and so forth.
<Esc> g	Moves to a specified line of the buffer. If this command is preceded by <Esc> n, this numerical argument is taken as specifying the line number. If no numerical argument is specified, the user will be prompted to provide a line number. If n is positive, the cursor is moved to the n th line from the top of the text. If n is negative, the n th line from the bottom of the text becomes the current line.

2.2 Deleting Text

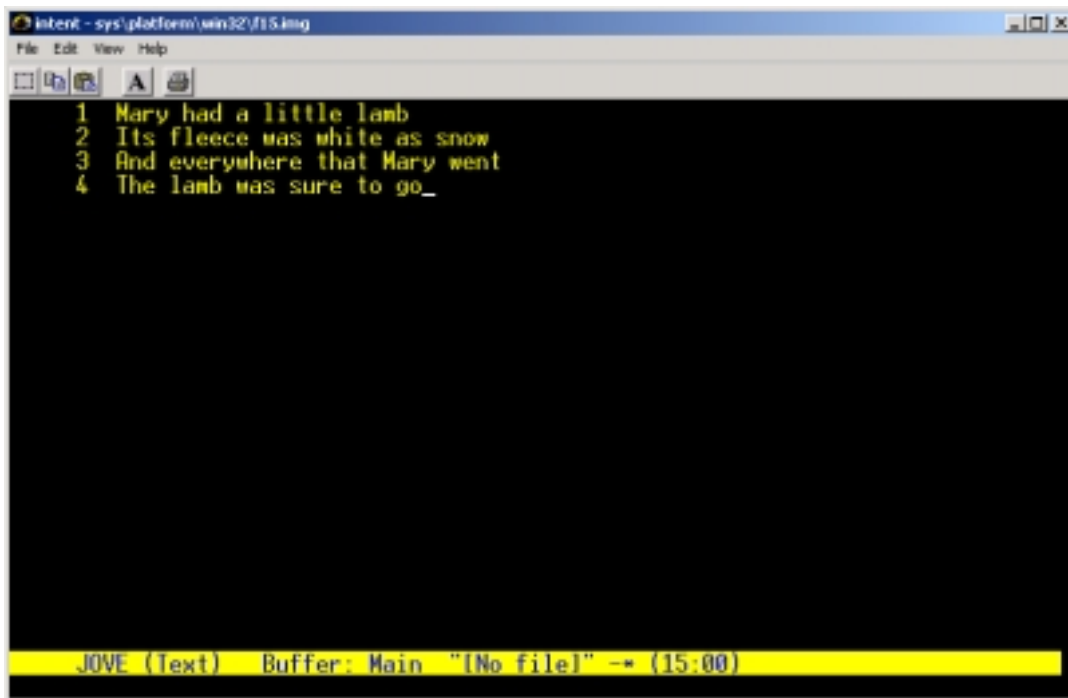
Getting Started With JOVE

<Back space>	Deletes the character to the left of the cursor.
<Ctrl-d>	Deletes the character on which the cursor is positioned.
<Esc> d	Kills the text between the cursor, or 'point,' and the end of the current or next word.
<Esc> <DELETE>	Kills the text between the cursor and the beginning of the current or previous word.
<Ctrl-k>	Kills (deletes) all characters to the end of the line.
<Esc> k	Kills (deletes) all characters to the end of the sentence.

Different commands exist for deleting blocks of text such as phrases and paragraphs. These are described in the section of "Manipulating Blocks of Text."

2.3 Numbering Lines

<Esc> x number- lines-in- window	This is a toggle command, allowing the user to turn on and off line numbering. When line numbering is turned on, the numbers should appear at the left side of each line. The editor does not consider the numbers to be part of the text.
---	--



2.4 Moving and Copying Text

<Ctrl-y>	Restores text previously removed by a "kill" command. "Kill" commands move text to a special buffer known as the "kill buffer." This is not displayed to the user, but its contents can be restored to the screen by typing <Ctrl-y>, which will cause the "killed" text to be inserted at the location of the cursor. The contents of the "kill buffer" will remain unchanged until another kill command is entered, therefore the text within may be "pasted" several times. To copy a section of text, the user should "kill" it, "restore" it to its old location, and then "restore" it a further time in a new location.
<Esc>y	JOVE remembers the last 10 sets of text killed by a "kill" command. Each of these can be retrieved by typing first <Ctrl-y>, then pressing <Esc> y one or more times until the desired text is displayed.

Getting Started With JOVE

2.5 Saving and Exiting

The text in the buffer is only a temporary copy of a file. In order to make it a permanent file, therefore, the contents of the buffer should be saved before exiting JOVE.

<Ctrl-x> <Ctrl-s>	Saves the contents of the temporary buffer into the file that is being 'visited.' Thus changes that have been made to the buffer are transferred into the file. If the changes are being saved from the MAIN buffer, the user will be prompted for the name of the file into which they should be saved. When the buffer contents have been saved, the editor will display the numbers of lines and characters saved.
<Ctrl-x> s	Saves the contents of the buffer into the associated file. If the contents of the buffer have not been modified since the last time that they were saved, then they will not be saved again. The editor will display a message to inform the user that no changes need to be written to file.
<Ctrl-x> <Ctrl-w>	Saves the contents of the buffer into a file other than that which has been opened. The user will be prompted to specify which file should receive the contents of the buffer. If the file already exists, the editor will notify the user of this, and ask whether it is acceptable for the existing contents of the file to be replaced.
<Esc> x write-region filename	Saves the contents of a defined block of text into a specified file. Details upon how to define blocks and regions may be found in the later section on "Manipulating Blocks of Text."
<Esc> x append-region filename	Saves a defined block into a specified file, without replacing the existing contents of the file. Instead, the new block is placed at the end of the text already contained.
<Ctrl-x> <Ctrl-c>	This leaves JOVE without saving the contents of the buffer. If changes have been made, the user will be notified, and given the option of aborting the exit.

2.6 Loading File Contents into Buffers

<Ctrl-x> <Ctrl-v> or <Ctrl-x> <Ctrl-r>	Switches files. This command clears all the text currently in the buffer, allowing the user to begin editing another file. After entering <Ctrl-x> <Ctrl-v>, the user will need to type the name of the next file to be opened, and then press <Enter>. If unsaved changes have been made to the buffer, the user will be asked whether these should be saved before the buffer contents are replaced. The user should respond by typing <y> for yes, or <n> for no.
<Ctrl-x> <Ctrl-f>	Loads the contents of a specified file into a new buffer. The buffer is then selected, and displayed upon the screen. If the file does not yet exist, JOVE will notify the user by printing "(New File)" in the mode line. If possible, the buffer is named after the filename.
<Ctrl-x> <Ctrl-i>	Inserts the contents of another file into the edit screen just after the cursor. The user will be prompted for the name of the file the contents of which are to be inserted. After the insertion, the cursor will be left at the start of the inserted file.

2.7 Paragraph and Line Commands

These commands may be used to redefine the alignment and layout of the paragraphs. In JOVE, a paragraph is a block comprised of one or more consecutive lines. A blank line will separate one paragraph from the next.

<code><Esc> x auto- fill-mode</code>	As the default, the editor will operate in "auto-fill mode." When using this mode, the cursor moves across the screen from left to right as text is typed into the buffer, until it reaches a pre-defined right margin. The cursor jumps down to the start of the new line only when the last line is "filled." If the editor is employing this mode, the word "Fill" will appear in the mode line. "Auto-fill mode" can be turned on or off by typing <code><Esc> x auto-fill-mode</code> . When this mode is turned off, the cursor does not start a new line unless <code><Enter></code> is pressed. Thus, <code><Enter></code> needs to be pressed twice to start a new paragraph.
<code><Esc> j</code>	Justifies the paragraph in which the cursor is positioned. If the left hand margin has already been adjusted, it will be necessary to type <code><Ctrl-u></code> first.
<code><Esc> x set right- margin n</code>	Adjusts the right margin that is used by the auto-fill mode, and while justifying text. <i>n</i> represents the column position, its default value being 70.
<code><Esc> x set left margin n</code>	Adjusts the left margin, setting it to the column position represented by <i>n</i> . The default value for this is 0. The left hand margin setting is needed for auto-fill.
<code><Esc> x auto- indent- mode</code>	Turns the "auto-indent mode" on or off.

2.8 Search and Replace

<code><Ctrl-s></code>	Searches the text for a specified string. After typing <code><Ctrl-s></code> , the user should type in the text to be sought by the editor, and then press <code><Enter></code> . The editor then searches forward from the location of the cursor, and displays the first occurrence of the string that it discovers. To display the next occurrence, the user should type <code><Ctrl-s></code> and then <code><Enter></code> . Unless otherwise specified, the search will be case sensitive. To make all ensuing searches case insensitive, it is necessary to type <code><Esc> x set case-ignore-search on</code> .
<code><Ctrl-r></code>	Searches the text for a string, as above, but scans backward through the text from the location of the cursor.
<code><Esc> q</code>	Searches the text for a specified string, and replaces it with another. The user must first type the string to be sought, then the replacement string. The editor will hunt out the first occurrence of the first string, and will ask the user whether to execute the replacement. Several responses may be made at this point. <code><y></code> executes the replacement. <code><n></code> leaves this occurrence unaltered. <code><u></code> undoes the changes made to the last occurrence. <code><p></code> replaces all occurrences of the first string with the second. <code><Enter></code> exits the 'search and replace mode.'

2.9 Manipulating Blocks of Text

A region of the text may be selected so that it can be deleted, moved or copied. To select a block, the cursor should be placed at either the beginning or end of the region, which should then be 'marked,'

Getting Started With JOVE

as shown below. The cursor should then be moved to the opposite extremity of the region, to define its other boundary. The region between the mark and the cursor is then selected.

<code><Esc> x</code> <code>set-mark</code>	Places a 'mark' at the current location of the cursor. The message "[Point pushed]" will be displayed on the screen, since the new mark will have been added to a ring of all previously set marks. As many as eight marks may exist on this ring. If the user wishes to choose a different ring from the mark as the 'current' mark, this can be achieved using the command <code>pop-mark</code> .
<code><Ctrl-w></code>	Deletes the selected region. The deleted text is stored in the kill buffer, and can be restored as normal.
<code><Esc> w</code>	Copies the selected region into the kill buffer. The block can be inserted into a specified location by typing <code><Ctrl-y></code> , as normal.
<code><Ctrl-x></code> <code><Ctrl-x></code>	Switches the locations of the cursor and the 'mark.' This command can be used repeatedly, and may be useful for jumping between distant locations in the text.

2.10 Cancelling a Command

<code><Ctrl-g></code>	Aborts a command that the user has started to issue.
-----------------------------	--

2.11 Repeating an Action

<code><Esc> n</code>	<p>It is often possible to cause a command to execute repeatedly by specifying a multiplier beforehand. To do this, type <code><Esc></code>, followed by the number of times the command is to execute. The command should then be issued as normal.</p> <p>It should be noted that if a "universal argument" such as this is used to repeat a screen moving command, the cursor scrolls by lines rather than screens. The command <code><Esc> 3 <Ctrl-V></code>, for example, moves the cursor forward three lines rather than three screens.</p>
----------------------------	--

2.12 Multiple Buffers

More than one buffer may be 'open' at a time, and several may be displayed on the screen simultaneously. The mode line will display the name of the buffer on the screen. If a new buffer is created, the previous buffer will continue to exist. Any new file will initially be empty. Files may be 'visited' as previously described, and their contents loaded into empty buffers. A different file may be edited in each buffer.

<code><Ctrl-x></code> <code>b</code>	Switches to a specified buffer. After entering this command, the user should type in the name of the buffer, followed by <code><Enter></code> . If this name matches that of an existing buffer, the editor will 'switch' to this buffer. If not, a new, empty buffer will be created and given the specified name.
<code><Ctrl-x></code> <code><Ctrl-b></code>	Lists the names of all the buffers currently in use. Any buffer containing unsaved changes will be marked with an asterisk. The user may switch to one of these buffers by typing <code><Ctrl-x> b</code> , as above, followed by the name of the buffer and <code><Enter></code> . If <code><Ctrl-x> b</code> and <code><Enter></code> are typed, without specifying a buffer, the editor will switch to the buffer most recently used. <code><Spacebar></code> can be typed to remove the list from the screen.

Getting Started With JOVE

2.13 Multiple Windows

JOVE is capable of displaying more than one window at a time. These can be used to show the same buffer, or as a means of viewing different buffers on the same screen.

<Ctrl-x> 2	Splits the current window into two. These will be displayed one above the other. This command may be used repeatedly, in order to split a window
<Ctrl-x> n	Moves the cursor to the 'next' window, so that it may be edited. The 'next' window will generally be the window below the 'current' window. Where the 'current' window is at the bottom of the screen, the 'next' window will be that at the top of the screen.
<Ctrl-x> p	Moves the cursor to the 'previous' window, so that it may be edited. The 'previous' window will generally be the window above the 'current,' or the bottom window if the 'current' window is at the top of the screen.
<Ctrl-x> d	This closes the current window.

2.14 Macros

It is possible to define macros within JOVE, using the following commands.

<Ctrl-x> (Starts defining a keyboard macro. After this has been typed, the editor will memorise all the user's keystrokes until the macro definition has been ended by typing <Ctrl-x>).
<Ctrl-x>)	Ends the definition of a keyboard macro.
<Ctrl-x> E	Executes the most recent keyboard macro.

2.15 Getting Help

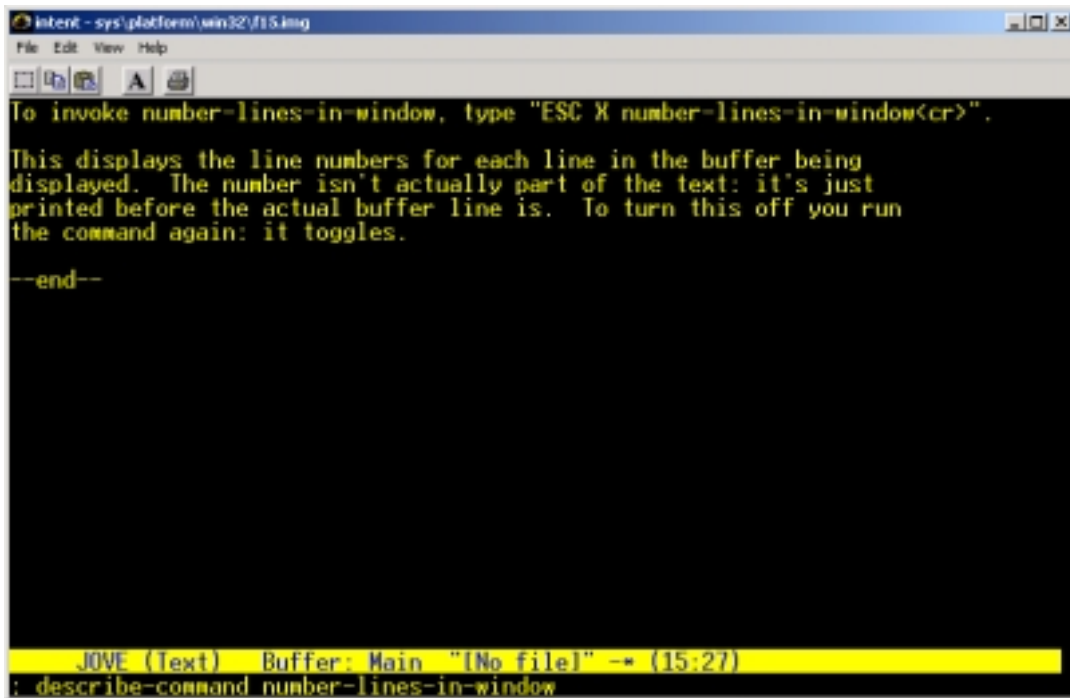
On-line help is provided with JOVE, and can be accessed through the command line. After typing <Esc> x, the user may type one of the following commands.

<i>apropos</i> <i>topic</i>	Displays a list of JOVE commands, variables and short-cut keys connected with a specific topic. After typing this, the user will be prompted for a keyword or phrase.
<i>describe-</i> <i>command</i> <i>command</i>	Provides a description of any specified JOVE command. Alternatively, <i>describe-command ?</i> can be typed to view a list of all commands. The <Spacebar> can be used to proceed through this list, and <Ctrl-g> to abandon it.
<i>describe-</i> <i>key</i> <i>key</i>	Provides the command line command that corresponds to a particular shortcut. If the user types <i>describe-key</i> , followed by a specific shortcut, the editor will display on the screen the name of the JOVE command to which it is 'bound.'
<i>describe-</i> <i>variable</i> <i>Variable</i>	Provides a description of a specified JOVE variable, such as case-ignore-search. Alternatively, <i>describe-variable ?</i> can be typed to view a list of all variables. The <Spacebar> can be used to proceed through this list, and <Ctrl-g> to abandon it.
Print <i>variable</i>	Displays the value of a specified JOVE variable. The command <Esc> x print right-margin, for example, would yield the value that defines the column position of the right margin.

It should be noted that the shortcut <Ctrl x> ? has the same effect as accessing the command line and typing *describe-key*. The shortcut <Esc> ? is "bound" to the command line command *describe command*.

Getting Started With JOVE

To leave any of the above help screens and return to a document, the user need only press the <Spacebar>. The help window may be scrolled by typing <Esc> <Ctrl-v>, and closed by typing <Ctrl-x> L.



© Tao Group Ltd or Tao Systems Ltd. 2000, 2001. All Rights Reserved.

Copyright in the software either belongs to Tao Group Ltd or Tao Systems Ltd. The software may not be used, sold, licensed, transferred, copied or reproduced in whole or in part or in any manner or form other than in accordance with the licence agreement provided with the software or otherwise without the prior written consent of either Tao Group Ltd or Tao Systems Ltd.

No part of this publication may be reproduced in any material form (including photocopying or storing it in any medium by electronic means and whether or not transiently or incidentally to some other use of this publication) without the written permission of the copyright owner.

*Elate®, intent® and the Tao logo are registered trademarks of Tao Group Ltd.
Digital Heaven™ is a trademark of Tao Group Ltd.
The rights of third party trademark owners are acknowledged.*