Bravo Manual

Table of Contents

c Xerox Corporation 1979

Preface

This manual describes the Bravo system for creating, reading and changing text documents on the Alto. It is supposed to be readable by people who do not have previous experience with computers. You should read the first four sections of the Non-Programmers Guide to the Alto before starting to read this manual.

You will find that things are a lot clearer (I hope) if you try to *learn by doing*. Try out the things described here as you read.

Material in small type, like this, deals with fine points and may be skipped on first, or even second, reading.

This manual is written on the assumption that you have the user profile, fonts and other Bravo-related material from the BASIC NON-PROGRAMMER'S DISK. If this is not the case, some of the things which depend on that stuff will not work the same way.

There is a one-page summary of Bravo at the end of this manual. It is intended as a memory-jogger, not as a complete specification of how all the commands work.

Bravo was designed by Butler Lampson and Charles Simonyi, and implemented mainly by Tom Malloy, with substantial contributions from Carol Hankins, Greg Kusnick, Kate Rosenbloom and Bob Shur.

1. Introduction

Bravo is the standard Alto system for creating, editing and printing documents containing text. It can handle formatted text, but it doesn't know how to handle pictures or drawings; for these you should use Draw, Markup or Sil.

When you start up Bravo (do it now, by saying <u>Bravo/e</u> <u>CR</u> to the Executive), you will see two *windows* on the screen, separated by a heavy horizontal bar. The top one contains three lines with some useful introductory information; it is called the *system window*. The bottom one contains a copy of the material you are reading, which was put there because of the "/e" you typed to the Executive. If you had omitted the "/e", as you do when using Bravo normally, the bottom window would be empty, except for a single triangular **endmark** which indicates the end of a document. In the bar separating the two windows is the name of the document in the lower window.

As you do things in Bravo, the first two lines of the system window will give you various useful pieces of information which may help you to understand what is going on and to decide what you should do next. Usually, the top line tells you what you can do next, and the second line tells you what you just did, and whether anything went wrong in doing it. *Make a habit of looking at these two lines while you are learning Bravo, and whenever you are unsure of what is happening*.

No matter what is going on in Bravo, you can stop it and get back to a neutral state by hitting the DEL key. You can leave Bravo and get back to the Executive by typing

> <u>Quit</u> <u>CR</u>

The characters which you type (Q and CR) are underlined in this example; the characters which are not underlined are typed by Bravo. This convention is used throughout the manual. Notice that you only type the first character of the Quit command; this is true for all the Bravo commands.

Each Bravo window (except the top one) contains a *document* which you can read and change. Usually you read the document from a file when you start Bravo, and write it back onto a file after you have finished changing it. Later, you will find out how to do this (section 2.3). It is possible to have several windows, each containing a document; this too is explained later on (section 4.2).

Bravo is controlled partly from the keyboard and partly from the *mouse*, the small white object with three black buttons which sits to the right of the keyboard. As you push the mouse around on your working surface, a *cursor* moves around on the screen. Pushing the mouse to the left moves the cursor to the left, pushing the mouse up (away from you) moves the cursor up; and so forth. You should practice moving the mouse around so that the cursor moves to various parts of the screen.

The three buttons on the mouse are called RED (the top or left-most one, depending on what kind of mouse you have), YELLOW (the middle one) and BLUE (the bottom or right-most one). They have different functions depending on where the cursor is on the screen and what shape it has. Don't push any buttons yet.

Mouse lore:

> You will find that the mouse works better if you hold it so that it bears some of the weight of your hand.

> If the cursor doesn't move smoothly when the mouse is moving, try turning the mouse upside down and spinning the ball in the middle with your finger until the cursor does move smoothly as the ball moves. If this doesn't help, your mouse is broken; get it fixed.

> You can pick the mouse up and move it over on your work surface if you find that it isn't positioned conveniently. For instance, if you find the mouse running into the keyboard when you try to move the cursor to the left edge of the screen, just pick the mouse up and set it down further to the right.

2. Basic features

This section describes the minimum set of things you have to know in order to do any useful work with Bravo. When you have finished this section, you can read the other parts of the manual as you need the information.

2.1 Moving around in a document

Move the cursor to the left edge of the screen and a little bit below the heavy black bar. Notice that it appears as a double-headed arrow. It will keep this shape as long as you stay near the left edge, in a region called the *scroll bar*. If you move it too far right, the shape will change. Keep the cursor in the scroll bar for the moment.

Now push down the RED (top or left) button and hold it down. Notice that the cursor changes to a heavy upward arrow. This indicates that when you let the button go, the line opposite the cursor will be moved to the top of the window. Try it. This is called *scrolling* the document up.

Next push down the BLUE (bottom or right) button and hold it down. Now the arrow points down, indicating that when you let the button go, the top line on the screen will be moved down to where the cursor is. Try it. This operation takes a few seconds, so don't get impatient. Practice scrolling the document up and down until you feel comfortable with it. It is useful to know that if you don't move the mouse, scrolling with RED and BLUE are symmetrical operations: one reverses the effect of the other.

You may have noticed that the text on the screen doesn't fill up the window, but that more text appears when you scroll up. The reason for this is that in addition to space on the screen, Bravo needs space inside itself (in the Alto's memory) to display lines of text on the screen. When a line has only a few characters, it doesn't take up much internal space, but when it runs all the way across the page, like the lines in this document, it takes a lot of internal space. When Bravo runs out of internal space, it stops displaying text and leaves the rest of the window blank. You can tell that there is more text in the document (i.e., that you aren't seeing the end), because when Bravo gets to the end it displays a triangular **endmark** as the very last thing to mark the end. If you don't see the **endmark** at the bottom of the displayed text, you can be sure that there is more text in the document which isn't being displayed.

If you keep the cursor in the scroll bar, near the left edge, and hold down YELLOW (the middle mouse button), you will see the cursor change into a striped right-pointing arrowhead. Think of it as a thumb, and the entire left edge of the window as the pages of a closed book, corresponding to your whole document (*not* just to what is displayed). If you stick the thumbnail into the book and flip it open, you will find yourself someplace in the book. If the thumb is near the middle, you will be about in the middle. If it is all the way at the top, you will be at the beginning; if all the way at the bottom, you will be at the end.

The tip of the arrowhead acts like the thumbnail, and letting go of YELLOW is like flipping open the book. You will also see another striped arrow, enclosed in a box. This one is called the *bookmark*; it points to your current location in the document. After you let up YELLOW, if you hold it down again without moving the mouse, the thumbnail and the bookmark should coincide exactly, making a solid arrowhead; this happens because the thumbing operation moved the document exactly to the place indicated by the thumbnail. To move forward a little, push the thumbnail down a little below the bookmark and thumb again; to move back, push the thumbnail up a little above the bookmark. To get to the beginning, push the thumbnail up until the arrowhead overlaps slightly the horizontal bar at the top of the window. Try thumbing your way through the document until you feel comfortable with it. Also try thumbing and then scrolling up and down.

2.2 Changing the text

In order to make a change in the text of your document, you have to:

> say where you want the change made, by making a *selection;*
> say what you want done, by giving a *command.*

You always make the selection first, then give the command. If you change your mind about where you want the change made, you can always make another selection. Making a selection is just like pointing with a pencil: it doesn't have any effect on the document. Only commands can change the document. You never have to worry about getting rid of a selection, since it never does any harm. If you make a selection, and then give a command that doesn't require any selection, that is perfectly all right; the needless selection will be ignored.

You make selections by pointing with the mouse and pushing one of the buttons. To try this out, move the cursor into the region of the screen where the text of the document is displayed. Notice that the cursor is displayed as an arrow which points up and slightly to the left. Point the arrow at a character (any character) in the document, and click RED. The character you pointed at should be underlined; if it is, you have just selected it. If it isn't, look nearby and see if some other character is underlined. If you find one, then that is the one Bravo thought you were pointing at. Experiment until you feel confident that you can point easily at characters.

You should note that each selection erases the previous one; there is only one selection at a time, and it is the most recent one. Also, you can make a selection at any time, except when you are in the middle of a command. Once you have started a command, you must either finish it normally, or abort it by striking DEL, before you can make another selection.

Something useful to know: if you hold RED down, you can move the cursor around and the selection will follow it. The selection won't freeze until you release RED (or move the cursor out of the text area). Try this too.

Now try a selection using YELLOW instead of RED. Notice that instead of underlining a character, Bravo now underlines a whole word. A word is defined as consecutive letters and digits, or consecutive punctuation characters. For convenience, apostrophe is counted as a letter. Also, a number containing a decimal point is a single word.

There is one more thing to learn about selecting text: how to select more than one character or one word. To do this, first select a character with RED. Then point to another character and click BLUE; Bravo will underline all the characters between the one you selected with RED and the one you pointed at with BLUE. This is called *extending* the selection. Try holding down BLUE and moving the cursor around. The selection will change continuously so that it includes the characters between the one you originally selected with RED and the one you are pointing at now. As before, when you let up the button, the selection will freeze. You can change the extension as many times as you want by using BLUE over and over; Bravo will remember the original selection you made with RED until you make another one.

Finally, try selecting a word with YELLOW and then using BLUE to extend the selection. Notice that the end of the selection will be a word also. To select the entire document, issue the <u>E</u>verything command.

Space, TAB and carriage return (CR) characters in the document simply appear as white space on the screen, just as they do on paper. You can, however, select them like any other characters. Try it. You will notice that not all the white space on the screen can be selected; in fact, the space on a line after a CR, and the space to the left of the left margin, cannot be selected. Bravo's handling of white space is discussed in detail in section 3.5.

Now that you know how to say where you want a change made, it's time to make a change. Select something (for instance, a word). Now type   D (for Delete). The word you selected is *deleted* from the document, and the selection moves over to the character after the original selection. The rest of the text is adjusted to make up for the deleted material; if necessary some words may be brought up from the next line to fill up the one which contained the deleted material.

You can *undo* the deletion by typing   U (for undo). Try it; you will see the stuff you deleted reappear, and it will be selected again, just as it was before you deleted it. Do several deletions, followed by undos, until you are sure you know what will happen. Try deleting larger pieces of text by extending your selections. Be sure not to move the selection between doing the Delete and the Undo.

Delete and Undo are *commands*. Like all Bravo commands, they are given by typing just the first letter of the command name. You can type the letter in either upper or lower case.

To add new text, select something in front of which you want the new text to go (if you want it to go at the very end of the document, you can select the **endmark**). Then type   I (for Insert). You will see that a blinking caret appears just before the selection. This marks the place where the new text will go. Anything you type will appear where the caret is, and as you type each character, the caret will move over to make room for it. Try typing a few characters, and notice that the rest of the text is automatically rearranged to make room for the new stuff.

If you strike the wrong key while typing, you can erase the mistake by striking the BS key (on the right side of the keyboard). You can erase as many typed characters as you like using BS. You can also use A^c (hold down the CTRL key and type A) to erase a character; it works just like BS, and may be more convenient to type with your left hand, if your right hand is on the mouse. To erase typing on a larger scale, you can use W^c (hold down the CTRL key and type W) to erase a word and its following spaces or punctuation characters. When you have typed as much as you care to, hit ESC to finish the insert. Notice that the caret disappears, and that the inserted material is selected. You can undo the insertion with   Undo. Then you can undo the undo and get the insertion back. Try it.

Sometimes it is more convenient to insert *after* a selection, rather than *before*. You can do this with the Append command (remember that you just type the A). Except for where the new material goes, Append is exactly like Insert.

If you want to change one word into another, or correct a typo, you have to delete some text and insert other text in its place. This can be done by a Delete followed by an Insert, but it is more convenient to use the   Replace command, which combines these two functions into one. Replace can also be undone.

Whenever Bravo first displays the blinking caret, you can insert a *copy* of some existing text rather than typing in new text. You do this by making another selection, called a *copy selection*, instead of typing. The copy selection is made exactly like an ordinary selection, and you can even use the scroll bar to move around in the document in order to find the text you want to copy. You can distinguish a copy selection from an ordinary one by its dotted underline, which contrasts with the solid underline of an ordinary selection.

You can change your copy selection as many times as you like. When you are satisfied with it, type ESC, and a copy of the copy selection will be inserted in place of the blinking caret. You can't do anything else while you are making a copy selection, except to scroll the document.

A copy selection can be used to move text from one place to another: first copy the text, and then delete the original.

There is one more useful thing to know about insertion. If you just type an ESC for an insertion, without making a copy selection or typing anything else, a copy of the last thing you inserted or deleted will be inserted. This is called repeating or *defaulting* an insertion; it is very convenient for inserting the same thing in several places, e.g., a dollar sign in front of several numbers. It also gives you another way to move text: first delete it, and then insert it in its new place by selecting the new place and typing   Insert followed by   ESC.

You now know all three ways of doing an insertion: *typing* the text, *selecting* some existing text to be copied, or *defaulting* the previous insertion by simply typing ESC. These three ways of inserting text can be used whenever a Bravo command needs some text. You will see many references to "inserting text" as you read on.

Before going on to learn anything more about Bravo, you should practice the Delete, Insert, Append and Replace commands, and copy selections, until they are quite familiar.

2.3 Filing a document

Whether you use Bravo simply to read or browse through a document, or to create or change it, you will need to fetch the document from a file before starting, and to file it away again afterwards if you have changed it. This section tells you how to do these things.

To fetch a document from a file, give the   Get command. You will see the blinking insertion caret appear in the heavy black bar above the window. Insert the text of the file name, usually by typing it in, and ending it with an ESC just as for any other insertion. The document will appear in the document window, and there will be a note in the system window telling you how long it is. A Get will erase the old contents of the window, if any.

To file a document away, give the   Put command, and type the file name as you did for Get. If the name you want is already in the black bar, you can just type ESC to default the name. It is also possible to edit the file name in the black bar, exactly like an ordinary document. Put always files away the entire document, regardless of what the selection is; when it is done, you will see a note which tells you how long it is. Bravo turns most of the screen black while executing a Put; this makes the Put run faster. Do not be alarmed at this.

Warning: If you make some changes to your document and then attempt to Quit from Bravo without having done a Put, Bravo will warn you that the document has not been filed and will ask you whether you still wish to Quit. If you want to save the document, strike DEL to cancel the Quit command, then file the document using Put. If you want to quit without saving the document, type Yes. If you do this, you will lose any changes you have made to the document. If this does happen to you, read section 4.3 on replaying to see if you can still be saved.

If you Get a document from a file and Put it back on the same file, Bravo will save the original on a *backup* file. Normally the backup file's name will be the name of the original file followed by a "$". The backup file is sometimes useful if you discover that some of the changes you made are not to your liking after all. If you have enabled version numbers at Install time (not recommended), the backup file will be the old version of the file from which you did the Get, and Bravo will make a new version for the Put (see section 8.7 of the Alto Non-programmer's Guide for a discussion of file versions).

You can do an "unformatted Get" with the Z^c command (type   Zc instead of   Get); this treats the formatting information at the end of each paragraph as ordinary text. The main use of Z^c is for patching up a file which has been damaged by hardware failure or cosmic rays. In particular, if Bravo refuses to Get the file because "End of file not in Bravo format", you can usually correct the problem by doing an unformatted Get of the file, deleting the last line or two, and Putting it back. Then Quit, restart Bravo and try again to Get the file.

2.4 Hardcopy

Printed copies of a document may be obtained using the Hardcopy command. Before using
Hardcopy for the *first* time, you must tell Bravo the name of the printer you intend to use regularly.
To do this, Get the file User.cm. In that document, you will find an entry that looks like this:

>   [HARDCOPY]
>   PREFERREDFORMAT: PRESS
>   PRESS: Name-of-your-Press-printer

Replace the words "Name-of-your-Press-printer" with the name of the printer you intend to use
(every printer has a registered name such as Clover, Menlo, or Daisy). Then Put the document,
Quit, and type   Bravo/i CR to the Alto Executive.

To print one copy of the document you are editing, simply give the   Hardcopy command followed
by CR. This will print the entire document, regardless of what the selection is. While doing the
hardcopy, Bravo displays in the cursor a count (modulo 10) of the number of pages it has
processed; hardcopy takes about 8 seconds per full page, like this one. After sending the document
to the printer, Bravo will report success. If there is a problem, Bravo will leave a note in the system
window. If the printer is not responding, Bravo will leave a note to that effect, and keep trying.
You can abort the Hardcopy by typing DEL, as always.

The hardcopy may fail for several reasons. If there is an EFTP error, trying again will usually work. If the problem is
that there is a character in your document which is in a font for which there is no printable representation, Bravo leaves
one of the offending characters selected, and puts it at the top of the screen. You can try again after modifying the
looks of the selected character. If you have a page with so many different fonts that it exceeds the capacity of the
printer, Bravo leaves the first character of the page selected and at the top of the window. There is no remedy for this
problem except to simplify the offending page. See section 4.6 for more information about fonts.

You may want more than one copy of a document. The Hardcopy command has an option called
Copies, which allows you to specify the number of copies you want; you type in the number, and it
will appear in the leftmost buffer in the system window, much like a file name. You must give the
Copies option right after the Hardcopy command, every time you want more than one copy.

If you compare the hardcopy of your document with Bravo's display, you will see that although the
text is identical, the hardcopy has more words on each line, so that the two versions look quite
different. In order to see a nearly exact facsimile of the hardcopy on the screen, you can give the
command

>   Look   hardcopy (note the lower-case h)

You are now in *hardcopy mode* on the screen. Until further instructed, Bravo will represent the
printed version of your document as faithfully as it can, by positioning each character on the screen
within one-half screen dot (about .007 inches) of its position in the final hardcopy. The screen
representation is 10% larger than the printed one. To turn off the hardcopy simulation, type

>   Look   Hardcopy (note the upper-case H)

You can edit normally in hardcopy mode. In fact, if your document contains tables whose
appearance is critical to you, it is advisable to stay in this mode, because in the normal mode text
will take up much more space on the screen than it will in the final hardcopy (if you have such tables,
you should also read section 3.5 on white space and tabs). In hardcopy mode it is also possible to see exactly
where lines will be broken, so that you can insert hyphens by hand if necessary.

Bravo provides a number of facilities for controlling page formatting, which you can read about in
section 3.6.

The Hardcopy command has options for printing on the Diablo printer, and for producing a Press file which can be
combined with drawings into a larger printable document, or sent to a file server for public distribution. These are
described in section 4.8.

2.5 Miscellaneous

As you edit, Bravo keeps track of the changes you make to the document. In doing this, Bravo consumes space in the Alto memory. During a long editing session, it is possible to consume all the available space, in which case Bravo will leave a warning note ("Core storage getting low") in the system window, and will refuse to execute any more editing commands. If this happens, you should Put your document onto a file immediately, and then Quit, restart Bravo, and Get the document back from the file. Now you can continue with another editing session.

When you have finished editing one document and have filed it away, you can Get another file, and continue working. It you are making extensive changes, however, it is better to Quit and restart Bravo when you start to work on a new document. If you do this, you are less likely to provoke a bug in Bravo, and you will be able to recover from a crash with the replay feature (section 4.3) much more quickly.

The maximum size of a Bravo document is 65,536 characters. Whenever Bravo Gets or Puts a document, it leaves a note telling you how long the document was. When your document has reached 65,536 characters, you won't be able to add any more text, and peculiar things may occur if you do try to add more text. It is a good idea to split the document into two parts well before this happens. To encourage you to do this, Bravo will flash the screen and display a warning message after every command if the length of the document exceeds 60,000 characters.

If you type a character which has no printable representation, Bravo will display it as a black rectangle. The best thing to do with such a character is to delete it.

Depending on exactly what Bravo is doing, the amount of text it can display on the screen will vary. You can always get the maximum amount of text displayed by doing a scrolling operation; if you scroll up with the cursor at the top of the scroll bar, the text won't move, and Bravo will just display as much more as it can. If you then give a command, some of the text may disappear from the screen, but you can always get it to reappear by doing another scrolling operation.

Bravo keeps copies within itself of information in your user profile (file User.cm; see section 4.6) and in various files on your disk: font files (named *.al and Fonts.Widths), and the files containing the Bravo system and its temporary storage (named Bravo.*). It refreshes these copies whenever you start it up with

>Bravo/i CR

This is called *initializing* Bravo. It is necessary to initialize whenever you get a new version of Bravo or the Alto Operating System and when you change your user profile or any font file. Initializing is just like starting Bravo up normally, except that it takes about a minute. *If you are in any doubt* about whether something has changed since the last time you initialized Bravo, or if your Bravo is crashing with messages which refer to disk or file errors, *you should initialize Bravo* by starting it with Bravo/i.

You now know enough to edit unformatted documents. Take a rest.

3. Formatting

This section describes the Bravo facilities for creating formatted text and pages. It you are not interested in formatting, you don't have to read it. If you are interested, be sure to read sections 3.3 and 3.4, where you will find a lot of good advice.

Bravo normally describes character sizes and distances on the page in *points*. A point is a unit of distance used in the printing industry; there are 72 points per inch. Thus 36 points is 1/2 inch, and 18 points is 1/4 inch. In many cases, you can also specify distances in inches or centimeters, as described in section 3.2.

3.1 Making pretty characters

Bravo allows you to say how you want your text printed: in italics or bold face, underlined, in various sizes and type styles, superscripted or subscripted, etc. You can change the way existing text is printed, or you can say how you want the characters to appear as you are typing them in. We will begin by describing how to change the looks of existing text.

First, select the text you want to mess with. Then give the   Look command. This command has a large number of options, each specified by a single character, which is sometimes followed by some additional information:

bold	SHIFT B to un-bold
italic	SHIFT I to un-italicize
to underline	SHIFT to remove the underline
_ to subscript (text down 4 pts)	Down 0 ESC to remove subscript or
SHIFT _ or ^ to superscript (up 4 pts)	superscript
0 to 9 to set the typeface	
visible to display spaces, tabs, and CRs	SHIFT V to stop this.

Down followed by a distance (see below) to move the text down that distance, relative to the baseline. Subscript is Down 4.

Up followed by a distance to move the text up. Superscript is Up 4.

CLR (the blank key to the right of BS on an Alto-I, or the key labelled BW on an Alto-II) to restore the standard looks: font 0; not bold, italic, underlined, visible, graphic, up, or down.

The typeface is usually called the *font*. For Bravo, each different size of the same style is a different font, but bold and italic are considered to be in the same font. The choice of fonts is specified by your *user profile* in a way which is described later (in section 4.6), but the standard choice provided on the basic non-programmer's disk is:

0	Times Roman, 10 pt. This is the standard font.
1	Times Roman, 8 pt.
2	XEROX logo (only the capital letters E O R and X)
3	Math, 10 pt. A large set of mathematical symbols. No bold or italics on hardcopy.
4	Greek, 10 pt. No bold or italics on hardcopy.
5	Times Roman, 12 pt.
6	Helvetica, 10 pt.
7	Helvetica, 8 pt.
8	Gacha, 10 pt. This is a fixed-pitch font.

9   Helvetica, 18 pt. The bold-face version of this font is especially good for making view-
         graphs.

You will find tables at the end of this manual which give the correspondence between ordinary
characters and the Math and Greek fonts, and some samples of the various fonts.

There is another Look option which is very convenient. It is   Look   Same, followed by a copy
selection. In this case, what is copied is the looks, rather than the characters. This is the way to
get one piece of text to print in the same style as another piece.

Like most commands, Look can be repeated with ESC. This is useful if you want to change the
looks of several pieces of text in the same way. You can also undo a Look with Undo.

You can find out what the looks of a character are by selecting it and giving the   Look ? command.
Bravo will tell you (in the system window) all the looks of the selected character. You may have to
scroll the system window up in order to see all the looks.

LOOKS DURING TYPING

When you start typing, the looks which will be attached to the characters you type are set to the
looks of
         the first character of the selection if the command is Insert or Replace;
         the last character of the selection if the command is Append;
         the standard looks otherwise.

To change the looks while you are typing text, use the CTRL key instead of the Look command:
hold down CTRL and type the look you want. The only things described above which you can't do
this way are Look Up and Down; you can get the standard superscript and substript offsets with ^
and _, though. To restore the standard looks, you can just strike the CLR key; it is not necessary to
use CTRL in this case.

3.2 Paragraphs

In addition to changing the looks of the characters, you can also change the shape of the text: the
margins, space between lines, justification, centering, etc. The Bravo facilities for doing this are
based on the idea of a *paragraph*.

A paragraph in Bravo is all the text between two CTRL-CR characters. You can tell when you have
one by selecting it. To do this, move the cursor into the *line bar*, which is between the scroll bar
on the far left, and the text area. You can tell that you are in the line bar, because the cursor will
appear as a rightward-pointing arrow. Once you are in the line bar, use the YELLOW button to
select a paragraph. Note that the cursor changes to a paragraph symbol; it keeps this shape as long
as the selection is a paragraph.

The CTRL-CR that ends a paragraph carries the paragraph looks described below. It can also carry character looks, and
if you are setting up a standard paragraph, it is a good idea to attach to its CTRL-CR the character looks which you
want as the standard ones for the paragraph. Thus, for example, the CTRL-CR for a standard heading like the one at
the start of this section would carry the italic look. Of course, this is just a convenience, and not essential; you can
always set the character looks during typein as described above, e.g. by typing   ic for italics.

If the text at the end of a paragraph is in a font smaller than the standard one, as this one is, the CTRL-CR ending the
paragraph should carry the same font looks. Otherwise, the inter-line spacing of the paragraph may appear uneven.

The YELLOW button selects exactly *one* paragraph, so by looking at what is underlined you can tell where the paragraph starts and ends. Note that the second CTRL-CR (the one that ends the paragraph) is counted as part of the paragraph; the first CTRL-CR is part of the previous paragraph. You can use BLUE to extend the selection to several paragraphs.

To merge two paragraphs into one, just delete the CTRL-CR that separates them. You will probably want to replace it with a couple of spaces, or maybe with an ordinary CR. To break one paragraph into two, insert a CTRL-CR; it is just like any other character, except that you can't backspace over it.

If you select a paragraph and then give an Append, Insert or Replace command, a blank paragraph with the same looks as the selected one will be created for you to type into.

To change the looks of a paragraph, you can use some more sub-cases of the Look command. Select the paragraph (or any text in it) first, and then say   Look, followed by:

    center; turns off justification   SHIFT C to stop centering

    justify (even right margin); turns off centering   SHIFT J to stop justifying

    nested to indent the whole paragraph (36 pts,   SHIFT N to un-indent
    or 1/2 inch, more)

    open up more white space in front of the   SHIFT O to close up the white space
    paragraph (12 pts, or 1/6 inch, more)

    q to open up half as much more white space in   SHIFT Q to close up the white space
    front of the paragraph as Open does (6 pts more)

All of these can be invoked during type-in; hold down the CTRL key and strike the appropriate key, just as you do for character looks.

In the following Look cases, *d* is a *distance* on the page, which can be specified in several different ways, as described below. Distances are measured from the left edge of the paper (except for Up and Down, which measure from the baseline of the line of text). These looks cannot be used during type-in.

    Left *d* to set the left margin. The default is 85 points, or about 1.2 inches from the left edge
    of the paper.

    First *d* to set the left margin of the first line. Use this to control indenting or un-indenting
    of the first line. A Look Left cancels a Look First, since it sets the left margin for all the lines of the
    paragraph.

    Paragraph *d* to set the left margin of all the lines except the first. A Look Left cancels a Look
    Paragraph, since it sets the left margin for all the lines of the paragraph.

    Right *d* to set the right margin. The default is 527 points. Since an 8.5" x 11" page is 612
    points wide, this results in 85 points, or 1.18", of white space on the right. Thus, the default
    margins center the text on the page.

    X *n* to set the space or *leading* between lines. The leading should be at least 1 point (as it is
    in this document) to avoid a squashed effect. If you want a less dense appearance, try larger
    leadings. The default is 6pt, which gives double spaced text.

    Y *n* to set the leading in front of the paragraph. The default is 12pt, which gives a blank
    line between paragraphs. Look open increases the paragraph leading by 12 pts, and Look q
    increases it by half that, or 6 pts. On hardcopy, both line and paragraph leading are
    suppressed for the first line of a page or column. Leading must be less than 64 points.

Here are the ways to specify the distance. Try them out until you are quite comfortable with them.

By typing a number in one of the following forms:

123 or 123pt	a distance in *points*. A point is a printer's unit equal to 1/72 of an inch. A number without a decimal point and with no explicit units is assumed to be in points.
1.71, 1.71in or 1.71"	a distance in inches. A number with a decimal point and no explicit units is assumed to be in inches.
4.34cm	a distance in centimeters.
a sequence of blanks	a distance equal to the width of that many blanks.

By typing a number *n*, as above, preceded by + or . The distance specified by *n* is added to, or subtracted from, the current value of the look being changed. Thus, to indent a paragraph by an additional one-half inch, type   Look   Left   +.5   ESC.

By using BLUE to point to a place on the screen. The horizontal position of the place you point at is displayed in the system window. If you hold down BLUE and move around, the displayed position is updated continuously.

By using RED to select a character. The horizontal position of the left edge of the character is displayed in the system window.

By typing \ (back-slash, not /), which displays a default value for the look being changed in the system window.

By just typing ESC, which uses the value already in the leftmost buffer of the system window.

You can select, point or default as many times as you want, just as with an ordinary copy selection. Then you can type a number, if you like. When the leftmost buffer in the system window has the value you want, type ESC to complete the command. Of course, if you get disgusted you can always type DEL to cancel the whole thing. Note that pointing is a convenient way to measure horizontal positions on the page.

Look All, followed by a copy selection, will copy all the paragraph looks of the paragraph in which the copy selection is made, to the paragraph containing the current selection. Note that Look All copies paragraph looks whereas Look Same copies character looks.

If a paragraph is selected (using YELLOW in the line bar; the cursor will be a paragraph symbol when a paragraph is selected), the   Look ? command will display the paragraph looks in the system window; if the selection is not a paragraph, the command displays character looks, as described in the previous section. You may have to scroll the system window to see all the information. Note that it appears in a buffer (see section 4.5) which is made current, and you can insert it into a document with a default insertion.

If you have a paragraph whose left margin is less than the default (normally 85 pts), any characters in the paragraph to the left of the default margin will fall off the left edge of the window and will not be displayed. Try setting a left margin to some values less than 85, and see how this works. You can change the setting of the left edge of the window, so as to make these characters visible on the screen, with the command

          Window Edge *d*   *d* is a distance, which must be typed in and cannot be obtained by pointing.

The distance *d* is the distance from the left edge of the page at which the left edge of the window should be set. It should be smaller than any paragraph left margin if you want to see all the characters on the left. For instance, if *d* is 0, the window edge will be at the paper edge; if the text has the usual 85 pt margin, this will result in 1.2" of white space in the window (in addition, of course, to the white space in the line and scroll bars). The default value for *d* is the default left margin.

HINTS

You can select several paragraphs (using BLUE to extend your selection) and apply the same Look command to all of them. You can change the looks of every paragraph in the document by doing an Everything to select the whole document before the Look. A Look command involving a distance of the form +*n* or *n* adds or subtracts *n* from the look value for each selected paragraph. Thus,   Look Left +5   ESC will indent each selected paragraph by five more points.

If you use several different formats (e.g., for section headings or for indented material) you can copy the formatting from an existing example of a particular style to a newly created one with Look All. Often it is convenient to put a set of sample paragraphs at the head of your document, each containing one line which explains what it is a sample of. Then you can split the window (as described in section 4.2) and have the samples readily available to copy from with Look All. This is highly preferable to entering all the new looks manually every time you switch to a new format.

An alternative technique for creating a new paragraph in a specific style is to select the paragraph before or after the point at which the new paragraph is to appear, then Append or Insert and make a copy selection of the desired sample paragraph. Now select the text of the newly-created paragraph, *not including the CTRL-CR at the end*, and Replace it with new text that you type in. This method copies both character looks and paragraph looks from the sample paragraph.

When you are setting up the format for a document, you should put a few blank paragraphs (just CTRL-CRs) at the end, and set the formatting on all of them to your standard format (it is convenient to do this by copying the formatting from a paragraph which already has your standard format). This might include indenting the first line of a paragraph, setting the leading, leaving space between paragraphs, justification, and even the font. Now when you add material to the document by inserting into one of these blank paragraphs, you will automatically pick up all of the formatting you have preset. As you type along, each time you use a CTRL-CR to start a new paragraph, it will acquire the same formatting as the old one.

3.3 Formatting style

This section is intended to provide you with some guidance in using all the different ways Bravo gives you for controlling the appearance of your document. Many of the rules are based on the customs of the printing industry. There are two advantages to following these customs:

  they are the result of many years of experimentation, during which many people have tried
  to find out what looks good on the page;
  readers are accustomed to seeing text presented in this style.

You will notice that some of the rules are contrary to the usual practice for preparing documents on a typewriter. There are good reasons for this: when you are printing with variable-pitch fonts, italics, boldface, justification, and precisely controlled leading, some of the things which work well for fixed-pitch, single-font documents are no longer appropriate.

EMPHASIS

Use italics for *emphasis* in text. You can also use boldface, but this is usually less desirable, and it is better to reserve boldface for words which play some special role, e.g., **begin** and **end** in computer programs. You should also use italics for the names of variables, e.g., "Suppose there are *n* items."

Don't use underlining for emphasis; it is not compatible with the use of italics and boldface. Use underlining only when you want a different *kind* of emphasis, e.g., to distinguish the characters a user types from the ones the machine types, as is done in this document.

Don't capitalize a whole word for emphasis. In fact, try not to capitalize a whole word at all; it usually looks terrible in a variable-pitch font because the capital letters are so much wider than the small ones. If you have words which you think should be set in capitals for some reason, try SMALL CAPITALS. In this example, the S and C were 10 point (font 0), the rest of the letters 8 point (font 1). Compare this with the appearance of FULLY CAPITALIZED words and you will see the point.

SECTION HEADINGS

In general, use left-justified rather than centered headings, and don't use all capitals, for the reasons just discussed. Here is a satisfactory list of styles for the headings of successively larger portions of your document:

smallest	*Italic*	18 pt paragraph leading (Look Y 18, or Look q if your standard leading is 12 pts).
next	**Bold**	24 pt paragraph leading (Look Y 24, or Look o if your standard leading is 12 pts).
largest	**12 pt bold**	36 pt paragraph leading (Look Y 36, or Look o twice if your standard leading is 12 pts).

Note that you can switch from the standard leading to the 1.5, 2 or 3 times standard leadings for headings during typein, using   oc and qc. For the largest units, you can center the heading and/or use all caps instead of, or as well as, switching to a 12 pt font. It is best not to have more than three levels of heading, but you can extend to four or five levels using these tricks. Helvetica 18 bold (font 9 bold) is sometimes nice for chapter or document titles.

Use Look Keep 80 (see section 3.6) on headings to make sure that the heading doesn't end up all by itself at the bottom of a page.

LEADING

The standard printing fonts are designed in such a way that they need some extra space between the lines to avoid a cramped appearance. You put this space in with Look X, and you should use 1 pt for ordinary single-spaced text. If you want a less dense appearance, experiment with more leading. For double-spacing of the text, try Look X 6 (the default).

Use double spacing (Look o) between paragraphs. When you have indented material which is fairly short, try 6 pt leading (Look q), as in the example two paragraphs back. *Don't* use extra carriage returns to get blank space between paragraphs. However, the maximum leading you can specify is 63 points; if you need more (e.g., to leave space for a figure) you will have to put in blank paragraphs.

Note that both line and paragraph leading are suppressed for the first line of a page or column. The height of a line of text (in points) is equal to the point size of the largest font used in the line, provided there are no characters which have been superscripted, subscripted or offset with Look Up or Look Down. If any character in the line is offset Up, the minimum line height, *including* leading, is given by the font size of the character, plus its offset; i.e., characters offset Up are allowed to eat into the leading. If a character is offset Down, the largest such offset must be added to obtain the line height; i.e., characters offset Down are *not* allowed to eat into the leading.

INDENTING

Use Look nested to indent material, and Look Nested to cancel the indentation. Note that this also works when you are typing in. For example, if you type

      CR^{c}n^c Here are three points: CR^{c}n^{c}First ..CR^{c}Second ..CR^{c}Third ..CR^{c}N^{c}Now we continue ...

the document will look like this:

      Here are three points:
           First ..
           Second ..
           Third ..
      Now we continue ..

Use Look First if you want to indent the first line of a paragraph, rather than tabs. When you have a list of items, it is often nice to *un*indent the first line by about 15 pts, especially if the items are numbered. For example:

1. This paragraph was formatted with Look Left 120, Look First 15, in order to make the number hang out to the left.

2. To get the first word of the first line to line up with the left margin on subsequent lines, set a tab stop at that point (see section 3.5).

3. The easiest way to specify the position of the tab stop is to select the first character of the second line, using RED.   In this case, of course, the stop is at 120.

4. Indended paragraphs sometimes look better balanced if the right margin is indented as well. Unfortunately, Look nested doesn't do this for you; you have to change the right margin yourself using Look Right *d*.

OFFSETS UP AND DOWN

Use the smallest offset you can get away with for subscripts and superscripts, since large offsets result in wide ugly spaces between the lines. The offset used by Look ^ (superscript) and Look _ (subscript) are defined in your user profile (see section 4.6); the standard profile sets it to 4 pts.

3.4 Forms

Although Bravo will let you begin with a completely empty window and start typing into it, this is a bad practice and should be avoided. Instead, you should start out by Getting a *template* or *form* which will guide you in constructing the document you want.

An obvious example is a memo form, and you will find one on the file Form.Memo. Start Bravo, and Get Form.Memo into the window. You will see that it has spaces for the sender, receiver, date and subject, and that these are filled in with words which indicate what should go there. To fill in the form, select each of these words, and Replace it with the proper text. Then do the same with the MEMOBODY. When you are done, you have a completed memo which you can file under a suitable name using Put. Be careful not to Put the document back onto the file from which the form came. The best way to avoid doing this is to edit the file name in the black bar above the document immediately after Getting the form.

Your disk comes equipped with a few forms; you can see their names by typing   <u>form.</u> TAB to the Executive. You should construct your own forms for other kinds of documents which you find yourself creating frequently. As you have seen in the description of Bravo's formatting features above, a form can contain a great deal of information in addition to standard text and spaces to be filled in. You will find that your life is easier and your work is more uniform and of higher quality if you use forms consistently, and take the trouble to carefully design a new one when necessary.

3.5 White space and tabs

When you type on a typewriter, you can get white space to appear between characters by typing spaces or TABs. You can get blank lines by typing carriage returns. In Bravo, you can do exactly the same things, with exactly the same results. Space, TAB and CR are characters which are in your document exactly like "a", "b" or "c". You can get Bravo to display them as special, visible characters by selecting the text in which you want to see them, and typing

        Look visible   (this must be a lower-case v).

To turn off the display and just see the usual white space, type

        Look Visible   (this must be an upper-case V).

Normally you don't have to type any CRs; Bravo will automatically end a line when there is no room for the next word. You can force a line to end by putting in a CR; this is appropriate when you want to control the layout of the text precisely, as in a table. Otherwise, don't put in CRs. You should use CTRL-CR to end a paragraph, as described in section 3.2.

Bravo allows you to set up to 14 tab stops, which are named by the digits 1-9 and the letters abcde. The tab stops are paragraph looks, just like the margins; hence they can be different for each paragraph. You can set a tab stop with the command

        Look   TAB *t d*

where *t* is a digit or one of the letters abcde, and *d* is a distance (see section 3.2).

When you strike the TAB key during typein, the caret moves to the next tab stop, just as it does on a typewriter, and a TAB character is added to the document. This TAB character is called a *plain-tab*, because it moves the caret to the next tab stop, not to a specific named tab stop.

For example, suppose you have a line like this:

        Column 1   Column 2   Column 3

Tab stops 1, 2, and 3 are at 180, 265 and 400 points, and there is a plain-tab between each digit and the following C. If you now append some **x**'s to the digit 1 to get past tab stop 1, the result will look like this:

        Column 1**xxxxxxx**   Column 2   Column 3

That is, the point to which a plain-tab jumps depends on the width of the preceding text. This can vary both when you change the text and when you switch to hardcopy mode; thus, the appearance of hardcopy may not match the screen if you are using plain-tabs.

You can turn a plain-tab into a *named-tab* by selecting it and issuing the command   Look **,** *t* (Look comma *t*), where *t* is the name of a tab stop. A named TAB character will always make the following character print at the correspondingly named tab stop. If printing has already passed that tab stop, Bravo will start a new line, and display a heavy black rectangle at the end of the previous line, to warn you that something is wrong.

To continue the above example, suppose you name the first TAB 1 and the second 2. Now the result will look like this:

        Column 1**xxxxxxx** *;*
                  Column 2   Column 3

When you switch from normal display mode to hardcopy mode, there will usually be more white space occupied by the TAB (perhaps enough to permit printing all the text on one line), but everything will continue to be positioned horizontally in exactly the same way.

You can find out the name of a tab stop by selecting it and giving the Look ? command.

Caution: the Look comma command should be applied only to TAB characters. If applied to a character other than TAB, it will invoke some unsupported features for color printing.

For compatibility with old versions of Bravo, and with the programmer-oriented tab conventions of the Alto and Maxc, you can set unnamed or *plain* tab stops spaced at equal intervals with the command

        <u>L</u>ook   <u>TAB</u> = *d*

where the distance *d* specifies the interval. If you don't set any tab stops, you get plain tab stops spaced at 36 pts (this parameter comes from the user profile, and can be changed; see section 4.6).

One final word about white space: Bravo has formatting features, described in the section on paragraphs above, which allow you to indent the first line of a paragraph, and to put blank space above a paragraph, without using spaces, TABs or extra CRs. It is good practice to use these features, since you can control the spacing much more precisely and don't have to worry about having extra characters cluttering up your document.

3.6 Page formatting

There are a number of features to help you in controlling the layout of your document on printed pages. Unlike the horizontal layout, the location of page breaks and the headings, page numbers etc. for the most part cannot be displayed on the screen. There is, however, a *page boundary* command which allows you to see on the screen where the page boundaries will appear in the hardcopy. The command is invoked by the LF key. It assumes that the first character of the current selection is the first character on a hardcopy page, and it moves the selection to the first line of the next page. By applying the page boundary command repeatedly, you can move through the document, page by page (or column by column, if your document profile specifies multiple columns; see below). Alternatively, if you know where one page break is (perhaps because of a control-L in the previous line; see below), you can start there. If you want to start at the beginning of the document, you can use the Everything command to make the first character of the document be the first character of the selection.

As a convenience, the page boundary command leaves the original selection at the top of one subwindow, and the first line of the next page as the third line of the next subwindow (which it creates if necessary). Among other things, this makes it easy to do some editing near the end of the page, and then reselect the beginning of the page and repeat the command. Try it.

Normally, Bravo will start a new page when it runs out of room on the current page, i.e., when the next line to be printed would intrude on the bottom margin, or at the beginning of a paragraph if the amount of space left before the botom margin is less than the paragraph's keep value. You can force a page break by including a L^c (control-L) in the text; the line containing the L^c will stay on the same page, but the next line will start a new page. This character is displayed as a lower-case L with an over-bar. You can't type it in simply by holding down CTRL and typing L, but instead you can type an L followed by S^c. You do this *during an insertion*, not as a command. The L^c is treated just like any other character during editing.

You can also force a *paragraph* to start a new page by giving it a keep property of 11". If you want to position the paragraph precisely on the new page, give it a vertical tab property as well.

You can exercise some control over where page breaks occur with the command

        <u>L</u>ook <u>K</u>eep *d*   *d* is a distance

This sets the paragraph property called *keep*, which has the following meaning. During hardcopy,

when printing of the paragraph is begun, the amount of space left on the page before the bottom margin must be at least the keep distance, or a new page will be started. For instance, by setting the keep of a heading to the total height of the heading (including its leading) plus the height of the first two lines of the next paragraph, plus the paragraph leading, you can ensure that the heading will never end up alone at the bottom of a page. Good values to use, with standard fonts and leading as in this document, are 40 pts on ordinary paragraphs and 80 pts on headings.

You can set the vertical position of a paragraph precisely on a page using the vertical tab property, which is set by the command

> <u>L</u>ook <u>Z</u> *d*   *d* is a distance.

When a paragraph with a vertical tab is printed, its upper edge (including leading, if any) will be positioned at the vertical tab value, measured from the *bottom* of the page (i.e., use 10.5" to put it .5" from the top). Unlike a horizontal tab, which may start a new line, a vertical tab never starts a new page; instead, it may cause overprinting. Vertical tabs are useful for positioning headings and footnotes, and for precisely aligning text to meet some physical constraint, such as a pre-printed form or a window envelope. The first line of a paragraph with a vertical tab will be printed on the current page, even if it runs into the bottom margin (but not if the paragraph also has a keep property which forces it off the page).

Vertical tab and keep properties are not visible on the screen, but you can always use Look ? to find out whether a paragraph has them, and what their values are.

Note that both line and paragraph leading are suppressed for the first line of a page of column. If you want white space in front of such a line *l*, you can use vertical tabs, or introduce a blank line in front of line *l*, and adjust the leading of *l* to compensate for the height of the blank line.

The remaining aspects of page formatting can be controlled by an optional *document profile* which you can put at the very beginning of the document. The document profile is a sequence of paragraphs, each of which must have the *profile* property. This property is set and cleared by a Look command:

> <u>L</u>ook <u>;</u> sets the profile property   <u>L</u>ook   SHIFT <u>;</u> clears it

A document profile has the following form (this one is the profile for this part of this manual):

> Page Numbers: Yes   X: 527   Y: .5"   First Page: 40   Not-on-first-page
> Private Data Stamp: No   X: 3.5"   Y: .6"
> Columns: 1   Edge Margin: .6"   Between Columns: .4"
> Margins: Top: 1.3"   Bottom: 1"   Binding: 5
> Line Numbers: No   Modulus: 5   Page-relative First Line: 1
> Odd Heading: Not-on-first-page
> > BRAVO MANUAL
> Even Heading:

Section 3: Formatting

Any of the lines may be omitted, and in general any of the fields on a line may be omitted. Fields on a line are separated by one or more spaces. Distances, shown in inches in the example, may be given in points or centimeters, as described in section 3.2. X coordinates are from the left edge of the paper, Y coordinates from the bottom; negative coordinates are measured from the right edge or top of an 8.5" x 11" page. Bravo's measurements on the page are exact to less than .01". Actual printers, however, can make errors in positioning the text on the page of as much as .25" in any direction. These errors do not affect the *relative* positions of characters (e.g., the length of a line cannot be affected) but they can cause the text to shift around on the page as a whole.

We now proceed to explain the various options.

PAGE NUMBERS

The coordinates of the page number are the coordinates of the upper right corner of the number. You can add Roman to the line if you want Roman numerals for your page numbers, and Uppercase if you want the Roman numerals in upper case. If Not-on-first-page is present, the page number is not printed on the first page of the document. If First Page is not specified, it is assumed to be 1, and Not-on-first-page is also assumed for both page numbers and heading; i.e., there will be no page number or heading on page 1 in the default case.

Each page number's looks are copied from the first character of the heading printed on the same page, if there is one, or from the first character of the document otherwise.

The coordinates of the private data stamp are for its upper right corner. Do not use a private data stamp without proper authorization. You will need to supply a password on each hardcopy to get the private data stamp applied; see your laboratory manager to learn the password if you have a legitimate need.

MARGINS

The top margin specifies the amount of white space at the top of the page. The bottom margin specifies the minimum amount of white space at the bottom of the page; a line will start a new page if any part of it intrudes into the bottom margin. Exception: if a paragraph has a vertical tab, its first line will be printed without regard to the bottom margin, and it may be positioned without any regard to the top margin.

If Binding appears, it is assumed that the pages are eventually to be printed on both sides of the paper, with odd-numbered pages on the right side of the resulting double spreads. Page numbers of even pages will be reflected left-to-right; in the example, even page numbers will have their upper *left* corner at X: .5" Y: .5". The binding distance is the amount of extra margin to be supplied on the inner side of the page, which abuts the binding. This amount is added to all the X coordinates on odd pages and subtracted from all the X coordinates on even pages. For example, if you want 98 pt (1.36") outside margins and 72 pt (1") inside margins, use a left margin (Look Left) of 85 pt (the default), and a right margin (Look Right) of 612 (8.5") 85=527 (also the default) to center the text on the page. Then use a Binding of 72 85= 13. In general, the rule is:

           Let d = (desired outside margin + desired inside margin)/2
           Look Left d
           Look Right   612 (8.5") d
           Binding: desired inside margin d.

This rule will lead to a negative binding if the inside margin is less than the outside margin; that is perfectly all right.

MULTIPLE-COLUMN PRINTING

The columns line is relevant only for multiple-column pages. It says that the hardcopy should have the specified number of columns, with the nominal edge margin (at both edges) specified (.6" in the example), and the amount of space between columns specified (.4" in the example). If the number of columns in the example is changed to 2, the nominal horizontal layout of an odd page will be:

           .6" edge margin; 3.45" text, column 1; .4" between columns; 3.45" text, column 2; .6" edge margin

for a total of 8.5". The text is centered on the page; if a Binding is specified, the text will be displaced in opposite directions on odd and even pages, just as for single-column text. The width of the text in the columns (3.45 in this example) is determined by subtracting all the other space from the 8.5" page width. If there are nc columns, the column width is

           col width = (8.5"   2*(edge margin)   (nc 1)*(between cols))/nc

The text width and position specified above is only nominal: the actual width and position is determined by the specified left and right margins in the following way. The first column is printed exactly as its left and right margins specify. The second column is moved right by (col width + between cols) from what its left margin specifies (i.e., that amount is added to all its X coordinates). This means, for example, that you can get a double-column page with some text at the top which runs all the way across by setting the right margin of the full-width text appropriately, and using a vertical tab to position the first paragraph of the second column below the full-width text. The appearance of the resulting page will be

<div align="center">

Full-width text ....
first-col text   second-col text

</div>

Note that to do this you must manually find the end of the first column (easily done using the page boundary command), and put a suitable vertical tab property on the first paragraph of the second column.

A consequence of this laissez-faire approach to column formatting is that you must supply the proper left and right margins yourself. To keep the text within the nominal boundaries defined above, the left margin should be greater than or equal to the edge margin specified in the document profile, and the right margin should be less than or equal to the edge margin plus the column width.

The edge margin specified in the example, which would be much too small for single-column pages, is good for double-column. It is also desirable to reduce the top and bottom margins when you are printing double-column, e.g., to .8" and .4" respectively.

When you are printing more than one column, a L^c in the document starts a new *column* rather than a new page. To start a new page, use two consecutive L^c characters.

LINE NUMBERS

If there is a line which says

Line Numbers: Yes Modulus: n Page-relative First Line: f

every nth line will be numbered, slightly to the left of the standard left margin. Thus, if n is 5, the numbers will be 5, 10, 15 .... If Page-relative appears, numbering starts over on each page; otherwise it continues throughout the document. If First Line appears, the first line is numbered f, and numbering continues from there; otherwise the first line is numbered 1.

HEADINGS

If a Heading line appears, it must be followed by a paragraph, also with the profile property, which is used as the heading on each page. This paragraph should have a vertical tab which positions it correctly (for example, 10.5" for the heading on this page) and appropriate margins, centering or whatever to produce the desired effects. It may have more than one line. It is also possible to have separate Odd Heading and Even Heading paragraphs. If Not-on-first-page is present, the heading will not be printed on the first page.

4. Other things

In this section you can learn about a wide variety of other useful things Bravo can do. They are described more-or-less in order of cost-effectiveness: the earlier ones will probably give you more payoff per unit of effort to learn them.

4.1 Some useful features

This section describes a number of features which are easy to learn and helpful to use. As always, it is a good idea to try them out as you read about them.

You can select entire lines of the document by moving the cursor into the *line bar*, which is to the left of the text area and to the right of the scroll bar. You can tell that you are in the line bar because the cursor will appear as a right-pointing arrow when it is in the line bar. To select the entire line pointed to by the cursor, use the RED mouse button. To extend the selection, use the BLUE button. Both of these work very much like selecting a character and extending. The YELLOW button selects a *paragraph*; you can read about paragraphs in section 3.2.

To put the current selection at the top of the screen, say   Normalize.

To insert the current *date and time* in front of the current selection, say   Time. Notice that it leaves just the time selected, so if you follow it immediately with a Delete, you will be left with just the date. To replace some text with the current date, select it and say Delete Time Delete; be sure you understand why this works.

You can *search* the document for the next occurrence of some text with the   Jump command. After you say Jump, you have to specify the text you want to search for, and you do this exactly the way you make an insertion: by typing it in (ending with ESC), by making a copy selection, or by typing ESC to default to the same text which was used for the last Jump (*not* the last insertion or deletion). Notice that if you type in text, it appears between the right-most set of curly brackets in the system window; this is called the *search key buffer*, and it normally contains the last text you searched for. However, the contents of this buffer are destroyed by the Look ? command and some of the Calculator commands. The search starts with the second line displayed in the window. If it succeeds, it brings the first occurrence of the text to the top of the window; if it fails, a note in the system window informs you. Jump does not affect the current selection. The search ignores the looks of the characters.

You can *substitute* one text for another using the   Substitute command. It will ask you (in the top window) for the information it needs. In looking for substitutions it will examine only the text in the current selection, so if you want to substitute throughout the document, do an Everything first; this will make the entire document the current selection. For reasons you don't want to know about, it is not a good idea to do a Substitute in which the old text contains a CR.

Most Bravo commands can be *repeated* by simply typing ESC in command mode. When you do this, Bravo uses the *current* selection, not the one which the previous command used. For example, you can append a carriage return after each of several words by selecting the first one and Appending after it, and then selecting successive words and simply typing ESC. Or, you can search through the document looking for occurrences of a word by Jumping to it once and then just typing ESC.

The Undo command will *undo* the action of most Bravo commands which change the document, provided you haven't moved the selection. You can only Undo the most recent command; it will still work if you have scrolled, however.

The (command will put parentheses around the current selection. You can put other kinds of brackets around the current selection with the commands   [, {, <, ' and   ".

The) command expands the current selection as little as possible to make it balanced with respect to parentheses. This is useful primarily while editing algebraic expressions or programs. For example, if the current selection is the underlined character in

$$X_{n+1} = ((aX_n) \text{ mod } (\quad \underline{m}+1) + c) \text{ mod } m$$

then one application of the   ) command will extend the selection thus:

$$X_{n+1} = ((aX_n) \text{ mod }\quad \underline{(m+1) + } c) \text{ mod } m$$

and another application will do this:

$$X_{n+1} = \quad \underline{((aX_n) \text{ mod } (m+1) + c) \text{ mod } m}$$

Again, the   ], }, and   > commands do similar things.

4.2 Windows

So far you have worked with a single document in a single window. Bravo will let you work on several documents at the same time, each in its own window. This is convenient if you want to compare two documents, or copy something from one into another, say from an address list into a letter. You can also have several subwindows looking into the same document, which is nice when you want to copy something from one part of the document to another, or to check something on another page without losing your place.

At the top of each window, separating it from the one above, is a heavy black bar. Inside this bar is the name of the file for the document in the window; this name is set by Get and used by Put. It can be edited like any other text. Subwindows, created only by the split operation described below, are separated by horizontal black lines. All the subwindows of a window are looking at the same document, although usually at different parts. If part of the document happens to be displayed in several subwindows, any changes to it will appear in all of them, and so will the selection underline or the insertion caret. Two different windows, on the other hand, are always looking at different documents, and no change to one window can affect the other. You can copy text freely from one document to another with a copy selection.

Some commands, like Jump, Everything, Get, Put and Hardcopy, work on the *current* window, which is the window containing the current selection.

There are two commands for windows, one for creating and re-arranging windows, and the other for destroying them. Each has three options, selected by the three mouse buttons.

To *create* a new window, type   <u>W</u>indow, move the cursor so that it marks the point where you want the new window boundary to be, and hold down BLUE. The new window will appear. As long as you keep BLUE down, you can move the cursor around and the top boundary of the new window will follow it. When you let go of the button, the boundary will freeze. Try it. The new window will be empty, but you can insert or Get into it.

To *split* a window and create a new subwindow, type   <u>W</u>indow, put the cursor where you want the split, and hold down YELLOW. The new boundary will appear, and it will follow the cursor until you release YELLOW. It is important to understand that after a split you have the *same* document in each subwindow. Scroll one of the subwindows so that some of the same text appears in both subwindows, and select some of the common text. Notice that the selection appears in both subwindows. If you make changes to the document, you will see them in both subwindows. This is very different from creating a new window and Getting the same file into it; that is equivalent to taking another copy out of a file cabinet.

To *move* a window or subwindow boundary, type   Window, put the cursor over the boundary you want to move, and hold down RED. The boundary will follow the cursor until you let go of RED.

You can get rid of a window or subwindow by typing   Kill, putting the cursor in the doomed window, and holding down RED or BLUE for about a second. RED will give the space of the window to the window above; BLUE will give it to the window below.   Kill YELLOW will clear the window; it is equivalent to Everything Delete (except that you can't Undo it).

If you attempt to Kill the only window showing a document that you have changed and haven't filed with Put, Bravo will pause with the message "Type CR to confirm". If you really want to lose the changes you have made to the document, type CR; otherwise, type DEL and then file the document using Put.

In summary:
	Window	Kill
RED	move boundary	give space to window above
YELLOW	split; new subwindow	clear
BLUE	new window	give space to window below

You can remove the *system window* from view by striking the unmarked key to the right of RETURN. This provides more space on the screen and in Alto memory for your document, and also makes subsequent commands execute slightly faster. You can restore the system window by striking the same key again. The system window is restored automatically when you make an error, and at certain other times when Bravo wants you to see what is in the window.

4.3 If Bravo breaks

When Bravo breaks or *crashes*, what usually happens is that Swat gets called; the manifestation is a couple of seconds of whirring from the disk, followed by a mostly blank display on the screen, with the words Swat version *xx* at the top. If this happens, look at the bottom of the screen, where there will be a more or less intelligible message. In some cases this message may describe a condition you can do something about, e.g., that your disk is full. Or it may inform you of a parity error; if this happens repeatedly, you should file an Alto trouble report to get your Alto repaired (see section 5.1 of the Alto Non-programmer's Guide). A third possibility is some fairly meaningless message describing an internal Bravo malfunction. In any case, after looking at the message you should type   Kc (if that doesn't work, boot the Alto). Then, if you want to recover your work, you can proceed as described below.

Bravo makes a record of everything you do during a session; the record is called the *transcript*. It is useful for three reasons:

   If Bravo crashes because it has a bug, the transcript can be used to report the problem to the people responsible for fixing bugs.

   If Bravo crashes because of a hardware failure of your Alto, a power failure, you accidentally pushing the boot button, or whatever, you can recover your work by *replaying* the transcript. In this case, the last few characters you typed may be lost.

   If you make a mistake, like deleting half of the document you have been editing for several hours, you can replay the transcript up to the error and then save the document.

You can do all these things using a system called BravoBug. Thus,

>BravoBug CR

will start replaying your transcript. As the replay proceeds, Bravo will report each command, just as it does when you type a command in the usual way. When it is finished, Bravo will say Ready, and you can go on editing. It is a good idea to save your work with a Put and start Bravo again.

Warning: you can only do a replay if you haven't started Bravo up again. Once you restart Bravo normally, your chance to replay is lost.

To *report* a bug in Bravo and then do a replay, type

>BravoBug/R CR

This will deliver to the Bravo maintainers copies of all the files involved in the bug, including the transcript, and then start Bravo to do the replay.

You can control the replay, step by step, as described in this paragraph. To stop a replay which is going on, type a space. As soon as the command currently being replayed is finished, Bravo will stop and tell you the *number* of the next command. At this point you can type

Quick to make typed-in text go in all at once during the replay, rather than one character at a time. This is faster, but you don't get to see what is going on. Quick is the normal mode.

Slow to make typed-in text go in one character at a time.

space to replay one more command. Note: only commands which change the text or windows are recorded in the transcript, not scrolling operations.

Proceed to continue replaying at full speed. You can stop the replay again at any point by typing a space.

Break before command *n* to make the next Proceed stop before command *n* (of course, it will still stop right away if you type a space). This is useful if you know that the first 50 commands are good, but want to proceed more cautiously after that. Note that after a Bravo crash, the Swat display usually tells you the number of the command during which the crash occurred.

Terminate CR to terminate the replay. After terminating, you can proceed to give ordinary Bravo commands. Don't do this unless you are sure that you want to stop replaying, since there is then no way to resume it.

You should try replaying a Bravo session and using these commands, so that you feel comfortable with them. You will then feel much more in control when you have a problem with Bravo or your Alto, or make a serious blunder while editing.

4.4 Arithmetic

Bravo incorporates a simple calculator, modeled after the Hewlett-Packard 35. The calculator has a *stack* with room for four numbers; while you are using it the top of the stack is displayed in the search key buffer, in the lower right corner of the system window.

To enter a number in your document onto the stack, select it and type   \ (enter). To add a number in your document to the top of the stack, select it and type   + (or   =, which is the lower-case character on the same key). Similarly, you can subtract with   , multiply with   * (or _), and divide with /. The   % command is just like *, except that it divides the result by 100. After any of these operations, the top of stack is the current buffer, which means that you can insert its contents in a document by defaulting the text of the insertion with ESC.

If you want to type in a number instead of selecting it, just type the number, and end it with one of the calculator commands. The number will appear in the middle buffer while it is being typed.

You can operate on the top two stack elements, rather than on the current selection and the stack, by prefixing the operation with the   C̲alculator command. Thus, to compute $(a + b) * (c + d)$, you

select *a*	\ (enter)	Stack: *a*	-	-	-
select *b*	+	Stack: *a+b*	-	-	-
select *c*	\	Stack: *c*	*a+b*	-	-
select *d*	+	Stack: *c+d*	*a+b*	-	-
C̲ *		Stack: $(a+b)*(c+d)$	-	-	-

This also works for enter: C̲alculator \ duplicates the top of the stack; if the stack was *a b c d*, it becomes *a a b c*.

There are a few more calculator commands which are occasionally useful:

~ exchanges the top two elements of the stack: *a b c d* becomes *b a c d*.

^ rotates the stack, bringing the second element to the top and the top to the bottom: *a b c d* becomes *b c d a*. Four repetitions of ^ leave the stack where it was.

C̲alculator *n̲* sets the number of digits after the decimal point to *n*. It is initialized to 2. All calculator arithmetic is rounded.

C̲alculator F̲ixed sets the display to fixed point (the normal mode).

C̲alculator S̲cientific sets the display to scientific notation.

C̲alculator E̲ngineering sets the display to engineering notation.

C̲alculator R̲adix *n̲* sets the radix. *n* can be a digit, or   B̲inary, O̲ctal, D̲ecimal or H̲exadecimal.

4.5 Other useful features

B̲UFFERS

The system window contains three pieces of text enclosed in curly brackets. These are called buffers, and they are used for a variety of purposes, some of which you have already encountered. The three buffers are numbered, as follows:

1 {last deletion}   2 {last insertion}   3 {search key}

One of the buffers is always marked with a "*"; that one is the *current* buffer, and its contents are usually what is inserted when you default a text insertion by simply typing E̲SC.

Commands which insert into buffers, like Jump and Substitute, default to the old contents of the buffer being loaded. Get and Put default to the file name already in the window.

The text in the buffers is always in visible mode, i.e., with spaces, T̲ABs and C̲Rs shown explicitly. Furthermore, T̲ABs and C̲Rs don't have their usual effect of leaving white space, because there is no room in the system window for these effects. Finally, if there is too much text to fit in the space allowed for the buffer on the screen, the middle of the text is replaced by an ellipsis (...).

You can force buffer *n* to be the current buffer with the command   B̲uffer *n* E̲SC. You can set the contents of buffer *n* with the command   B̲uffer *n̲* followed by typing or a copy selection.

PARTIAL SUBSTITUTION

If you want to substitute "that" for "this" you can use the Substitute command described in section 4.5. If you want to change *some* of the occurrences of "this" to "that", however, it is useful to know about the Find and Yes commands.

Find is exactly like Jump, except that

  the search starts from the end of the current selection, not from the top of the display;

  the occurrence of the key which is found becomes the current selection (Jump leaves the selection unchanged).

Yes is equivalent to   Replace ESC Find ESC. So, to change some "this"es to "that"s, proceed as follows:

  select something before the first place you want to start looking;

  Find   this ESC;

  ESC to repeat the Find until you get the one you want to change;

  Replace   that ESC;

  Find   ESC;

  Now at each point type   ESC (i.e., repeat the Find) to make no change and go on to the next "this", or   Yes to make the same change you made last time and then go on.

CONTROL AND SPECIAL CHARACTERS

Bravo normally displays a control character as the corresponding lower-case letter (or whatever) with an overbar. If you turn on   Look graphic it will try to display the control character from the font (if there is something in the font for it; otherwise it will display a black rectangle). This is a character Look, just like Look Visible.

You can't type a control character in directly, but you can type the corresponding regular character, *followed* by a S^c, which converts the preceding regular character into a control character.

Certain font families (notably TimesRoman and Helvetica) contain extra characters, such as accents, ligatures, and several sizes of dashes and spaces, which are useful in creating very high-quality documents. These may be entered as control characters in graphic mode (use   Look graphic). The correspondence between control characters and their printed representations is as follows.

  Accents. Each of these has a width of zero so should be typed in immediately *before* the character above which it is to appear.

d s^c	example:	o
e s^c		o
k s^c		o
p s^c		o

  Ligatures:

f s^c	replaces:	f f
q s^c		f f i
r s^c		f f l
t s^c		f i
u s^c		f l

Quotation marks:

	g s^c	open single quote
,	,	close single quote (standard apostrophe)

Note that you can make open and close   double quotes'' by using pairs of the corresponding single quotes'.

Spaces and dashes:

| |	space bar	letter space; here are 10 of them: | |
| |	y s^c	figure space: | |
| |	o s^c	em space: | |
-	-	hyphen: ----------
	v s^c	en dash:
	x s^c	minus sign:
	s s^c	em dash:

Miscellaneous:

	b s^c
	c s^c
	h s^c

At the time this manual was prepared, the special characters had not yet been incorporated into the *.al display fonts. Until this has been done, these characters will all appear as black rectangles on the screen. To find out which control character one of these black rectangles corresponds to, turn off its graphic look.

4.6 The user profile and fonts

The file User.cm is your *user profile*, which contains information for various systems about how you want them to be set up for your use. The information for each system starts with the name of the system in brackets, e.g. [BRAVO]. Then follow a series of lines of the form

> label: information

Bravo currently accepts three kinds of information in the user profile: initial and quit *macros*, described in the next section, and font definitions and default parameter settings, which are discussed here. Look at your User.cm file now, to see how this works.

Each line of font definition tells

> the *number* of the font (0 to 9)

> the *name* and *size* (in points) of the hardcopy font

> the name and size of one or two screen fonts which can be used to represent that hardcopy font on the screen. Bravo will use the first one normally but the second one (if present) in hardcopy mode.

For example, the line

> FONT:0 TIMESROMAN 10 TIMESROMAN 12 TIMESROMAN 10

says that font 0 is to print as 10 pt Times Roman when you generate hardcopy.

The rest of the line says that font 0 can be represented on the screen by the screen fonts stored on files TimesRoman12.al and TimesRoman10.al, which must be present on your disk. Bravo will use the 12 pt version in normal display mode and the 10 pt version in hardcopy mode. The extension "al" is used for screen fonts. There are no files for the bold or italic versions of screen fonts,

because Bravo can construct them from the regular version.

In addition to all the *.al files, you must also have a file called Fonts.widths, which contains information about the widths of all the characters in the hardcopy fonts.

Warning: A Bravo document file does not contain the association between the font numbers and names used to create that document. If you modify the font definitions in your user profile, you may encounter problems when you attempt to share Bravo documents with other people.

The user profile also contains settings for

default left and right margins;

tab interval;

Look nested parameters;

default line leading (used by Look X) and paragraph leading (used by Look Open, Look Q and the default for Look Y);

standard offsets (used by Look ^, Look _, and the default for Look Up and Look Down).

You can change all these settings by editing User.cm in a way which should be obvious; after doing this, be sure to initialize Bravo. Except for the default margins, all these settings affect only the process of editing the document, and not the document itself. In other words, once a Look nested, superscript or whatever has been done, the margins, offset, etc., have been set in the document and cannot be affected by subsequent changes in the user profile.

There is also a [HARDCOPY] section in the user profile that is used both by Bravo and by other subsystems that generate hardcopy. This contains entries for

preferred printing format (Press is the only reasonable choice at present);

name of your Press printer;

the name to be inserted in the "Printed by" field on the break page of the printout (an occurrence of "$" is replaced by the name by which you are known to the Alto Executive).

4.7 Startup and Quit macros

This section is only for programmers, and is not recommended even for them.

You can put into your user profile sequences of Bravo commands, called macros, to be executed automatically when you start up or quit from Bravo. Each macro is named by a letter. Startup macro *x* will be executed if you start Bravo with

>Bravo/x . . .   CR

Quit macro *x* will be executed if you type an "x" instead of CR after typing Quit.

You can see the format of a macro definition by looking at your user profile. The command sequences are just like those which Bravo writes into the transcript (the file Bravo.ts), and can be constructed by actually executing the desired sequence of commands, and then copying Bravo.ts to another file and copying the sequence out of that file. There are two exceptions:

* in place of a selection (which looks like {6,2,123,648}) means the current selection;

@n in place of typed-in text (which looks like 'text{) means the *n*th *parameter*. For startup macros the parameters are strings on the command line separated by blanks; the first one after Bravo/x is numbered 1. For quit macros, parameters 1, 2 and 3 are the three buffers, 4 is the file name for the first window, 5 the file name for the second window, etc.

4.8 Press and Diablo hardcopy

To make a Press file on your disk instead of sending your hardcopy directly to a printer, use the File option of the Hardcopy command. It will ask you for the name of the file; the recommended name is the name of your document, with the extension replaced by "Press". Once you have made this file, you may print it using the EmPress subsystem, store it on a file server to make it available to other people (see sections 6 and 7.4 of the Alto Non-programmer's Guide), or give it to PressEdit to combine it with other files into a large document (see section 7.6 of the Guide). *Warning*: the Press file is typically 50% larger than the document file; be sure you have enough room on your disk.

To send hardcopy to some printer other than your regular one (specified in User.cm), use the Hardcopy option

> @ *name*   ESC

where *name* identifies the printer you wish to use.

For both Press and Diablo printing, you can start printing at a specified page number with the Hardcopy option

> Start at page *n*

This is mainly useful for Diablo printing. There will usually be a substantial delay while Bravo figures out where page *n* starts.

To print on the Diablo printer, you can use the   Diablo option to the Hardcopy command. Before doing this, be sure a printer is plugged into your Alto; when you plug or unplug it, *turn the Alto power off first*. This option has an array of sub-options, which you can invoke when it pauses before printing each page. At the pause, the system window says "Ready to print page *n*", and the beginning of the text for page *n* is displayed in the document window (if there is room in Bravo's memory). You can then say:

   Repeat last page to prepare to reprint the previous page, instead of the page which was going to be printed next.

   Continuous print, to suppress the pause after each page; this is useful if you have continuous forms in your printer. You can still stop the printing by typing space during printing, as described below.

   space to start printing the page which Bravo says it is ready to print.

During printing, you can abort printing of the current page at any time by typing a space. You can then use the options just described to restart the current page (with space) or reprint the previous page (with Repeat). If you want to start at another page, use DEL to leave the Hardcopy command, give another Hardcopy command, and use the Start at page *n* option.

There is a command   Look Magnify Diablo that displays the document as closely as possible to the way it will appear when printed on the Diablo printer. Look Hardcopy terminates this mode.

THE SCREEN

```
What you can do next
What you did last
1{delete} 2{insert} 3{search key}
file name
S   L
C   I
R   N
O   E
L
L   B        TEXT
    A        AREA
B   R
A
R
```

SYSTEM
WINDOW

DOCUMENT
WINDOW

THE MOUSE

RED

YELLOW

BLUE

WHAT MOUSE BUTTONS DO

	Scroll bar	Line bar	Text area
RED	Scroll up	Select line	Select character
YELLOW	Thumb	Select paragraph (Sets cursor to ‖)	Select word
BLUE	Scroll down	Extend	Extend

The CLR key is the blank key to the right of BS on Alto-I
   or the key labelled BW on Alto-II.
The blank key to the right of RETURN removes or restores
   the system window.

COMMANDS

CHANGING TEXT

Delete selection
Insert *text* before selection   (These make a new
Append *text* after selection   paragraph if a paragraph
Replace selection by *text*   is selected.)

During type in: Backspace character with BS or ctrl-A
                   Backspace word with ctrl-W
                   Separate paragraphs with ctrl-CR
                   Enter ctrl- *char* as *char* ctrl-S

text may be typed or selected. ESC gets the current buffer. Buffer *n* ESC makes buffer *n* current.

FILING

Get to read in a file   Hardcopy to print. Options: Copies *n* Start on page *n*   CR to do it
ctrl-Z for unformatted text   File *name* .press (no printing)   Diablo output, with sub-options: Space to start or abort
Put to write out a file   @ printer-name   Black (ignore color)   Continuous printing
File name goes into bar above window. ESC gets you one already there. Can edit file name like any other text.   Reprint last page

MISCELL- ANEOUS

Undo works on most recent command.
ESC repeats most recent command, using current selection.
Everything selects whole document.
Normalize moves current selection to top of window.
([{ < ’ " put indicated brackets around current selection.
)] } > extend current selection to closest matching brackets.
LF takes current selection as top of page, and moves selection to top of next page.

DEL cancels what you are doing.
Quit CR exits from Bravo
Time inserts current date and time.
Buffer 123 *text* sets buffer and makes it current.
   ESC instead of *text* just makes it current.

LOOKS

Standard Fonts

0 Times Roman 10
1 Times Roman 8
2 Logo 24
3 Math 10
4 Hippo 10
5 Times Roman 12
6 Helvetica 10
7 Helvetica 8
8 Gacha 10
9 Helvetica 18

There are samples
   in the manual

Basic looks.   Mostly, SHIFT means NOT. Thus, Look b turns on bold, Look B turns it off.
                 During type-in, use ctrl-b for bold, etc. In command mode, give the Look command, then the letter alone (no ESC).

Text: bold   - underline (highlight if color)   Paragraphs: centered   justified
         italic   _/^ sub/superscript   q - add 6 pts   open - add 12 pts space before paragraph
         visible   0-9 font   nested (indent)
         graphic   CLR to reset   ; document profile

Look has more cases, which take a distance as parameter. These cannot be used during type-in.
Text: Up raises text; default 4 pts.   Paragraphs: Left margin; default 85 pt
         Down lowers text; default 4 pts.   F - left margin of first line
         , *n* names a tab (*n* = 1-9, a-e)   P - left margin of other lines
         . *c* sets color (*c* = Black, Cyan, Green, Magenta, Red, Violet, Yellow)   Right margin: default 527 pt
Distances (measured from left or bottom page edge):   X - space between lines; default 6 pt
         As a number: in points (72/inch) - 123 or 123 pt.   Y - space between paragraphs; default 12 pt
                         in inches - 4.5 or 4.5" or 4.5 in   Z - vertical tab; default none
                         in cm - 4.5 cm   Keep on current page; default 0; 11" forces
                         in blank widths - type that many blanks   new page
         As an increment to current value: + or - followed by a number as above   TAB n d sets tab stop n (n=1-9, a-e); default not set
         As the left edge of a character: select with RED   TAB = d sets even tab stops, default 36 pt
         As a position on the screen: point with BLUE
         Defaulted to a standard value: type \.

Look Same *selection*   sets all text looks of current selection to be the same as those of *selection*
Look All *selection*   sets all paragraph looks of current selection to be the same as those of *selection*
Look ? displays looks of selected text or paragraph in buffer 3.
Look hardcopy makes screen match hard-copy.   Look Hardcopy clears it.   Look Magnify Diablo matches Diablo hardcopy

PAGE FORMATTING

The following can be in the document profile; it must be the first paragraph, and have the profile property (Look;):
         Private Data Stamp: Yes/No X:d Y:d   d is a distance: use - to measure from top or right
         Columns:n Edge Margin:d Between Columns:d   ctrl-L causes page (or col) break after current line
         Margins: Top:d Bottom:d Binding:d   Use two ctrl-Ls for page break with multiple cols
         Line Numbers: Yes/No Modulus:n Page-relative First Line:n   Also note Look Keep and Look Z (vertical tab)
         Page Numbers: Yes/No X:d Y:d First Page:n Roman Uppercase Not-on-first-page
         Heading or Odd/Even Heading: Not-on-first-page followed by a heading paragraph, also with profile property

WINDOWS

Window RED   to move boundary   Kill   RED   to merge with window above
        YELLOW   to split (make new subwindow)   YELLOW to erase contents, leave the window
        BLUE   to make new window   BLUE   to merge with window below
Window Edge *distance* sets the left edge of the window at the specified point on the page; default is 85 pt.

SEARCHES

Jump *text*   starts search at second line, doesn't move selection
Find   *text*   starts search after current selection, moves selection to the string found
Substitute *text* for *text* - works on current selection. Usually you want to do Everything first.
Yes is equivalent to Replace ESC Find ESC; use Yes when you want to confirm substitutions.

REPLAY

>Bravobug will start a replay.   >Bravobug/R will report a bug to the Bravo maintainers first, then replay.
Space will stop the replay and show you the number of the next command. You can then type:
         Slow to slow down typein   Space to replay one more command   Break n to stop before command n
         Quick to speed it up   Proceed to continue replay full speed   Terminate CR to stop replaying

ARITHMETIC

Operators are += - *_ % / \(enter). An operator as a command combines the selection with the top of stack.
You can also type a number, followed by an operator. The stack is just like the one in a Hewlett-Packard calculator.
Also: ~ exchanges x and y   Calculator, with options: Fixed   operator to operate on x and y
         ^ rotates the stack   Scientific 0-9 to set digits after decimal pt
                                                                        Engineering Radix n to set the radix (n = H for hex)