Handling Justified Text

With System 7.0, the **Script Manager** provides several changes that are needed for handling fully justified text in the following list of **Script Manager** routines. These routines assume that a **slop value** for the line-the difference between the desired width and the actual width before justification-is to be distributed among the style runs on a line and among the words and characters within a style run. The actual width before justification is the sum of values returned by the **TextWidth** function for each style run on the line.

Here is a summary of how the justification routines available prior to system software version 7.0 work. (For details on these routines, see *Macintosh Worldwide Development: Guide to System Software*.)

- **PortionText** lets you determine how to distribute the slop value for the line among the style runs on the line.
- **DrawJust** allows you to draw a style run on a line and to provide a slop value for the style run.
- **MeasureJust** fills an array that specifies, for each character in a style run, the width from the beginning of the style run through that character.
- **Char2Pixel** converts a character offset in a style run to a pixel width in the style run.
- **Pixel2Char** converts a pixel width in a style run to a character offset in the style run.

The Roman Script System versions of the original justification routines did not handle intercharacter spacing and did not provide enough information for these routines to supply proper intercharacter spacing between style runs when there are multiple style runs (that is, more than one font, size, or **QuickDraw** style) on a line.

In System 7.0, all of these routines handle intercharacter spacing properly in all scripts and on all ports. When relevant, the amount of intercharacter spacing can be controlled using the **Script Manager** smgrCharPortion global variable, which can be set with the **SetEnvirons** function.

Version 7.0 also provides a second interface for each of the existing routines. Each new interface name begins with *N* (for example, **NPortionText**), and each interface contains additional parameters that specify the visual position of the style run in the line and the desired scaling factors. The scaling factors are supplied as numerator and denominator parameters in a manner identical to the **QuickDraw** **StdText** procedure. The slop argument in these routines is a signed value that specifies the number of screen pixels by which the style run should be extended (or shrunk, if the value is negative) after the numerator-denominator scaling has been applied.

Other improvements in the new routines for handling justified text include the following:

- For future use, parameters that specify a character position or length change from short to long; parameters that specify a pixel width change from short to fixed. However, some routines currently only use the short part of a long parameter.

- **NPixel2Char** includes a parameter *widthRemaining*, which is the adress of a Fixed). If the *pixelWidth* parameter that is passed into **NPixel2Char** is greater than the width of the text specified by the *textBuf* and *textLen* parameters, the amount of excess width is returned in the *widthRemaining* parameter; otherwise, *widthRemaining* receives -1.

- A new picture opcode saves the line layout information needed for these routines. See **Color QuickDraw** for a discussion of the new picture opcode.