Text Edit record Types

The TERec structure itself has not changed, but several fields have new meaning, in an upwardly-compatible sort of way:

- When the txSize field is set to -1, the changes have been implemented and the record is "style-aware".

- When lineHeight and fontAscent are -1, **TextEdit** performs automatic line-height calculations. The height of each line is stored in an array of type LHTable. You may override the "floating" height of any line by setting the high bit of its LHElement structure.

- The 4 bytes starting at txFont (offset 74) contain a handle to a TEStyleRec structure containing all the style-related information (or handles leading the the various sub structures).

Thus, when you get a TEHandle, just examine the txSize field to see if the record is "style aware" and take appropriate action.

The Style Record

The TEStyleRec structure is the starting point for all the structures associated with styles. See that structure for details, but in general, the system works like this: Each unique set of attributes has an STElement structure and there is an array of these elements, called the *style table*.

TextEdit maintains a linear array identifying *style runs*; a run is defined by a starting point (an offset into the data at TERec.hText), and an index into the style table. As text is inserted or deleted, these style runs are updated to reflect the changes. When a style is applied to text, the resulting combination is compared to other elements of the style table and if it doesn't exist, a new style is added to the style table. Then style runs are updated to reflect the changes.

To keep track of the now-variable height of lines, **TextEdit** maintains a linear array of line-height information (see LHTable). This array parallels the elements of the TERec.lineStarts array (the variable-length array found at the end of the edit record). For instance, TERec.lineStarts**[***n*] is the offset of the first character of line *n* and LHTable[*n*] contains the height of that line. The line heights are calculated automatically but you may force a recalculation via **TECalText**.

The Style Scrap

The StScrpRec structure is a variable-length structure defining runs of styles that apply to any unformatted text. When you use **TECut**, **TECopy**, or **GetStylScrap**, **TextEdit** builds up one of these structures. Unlike the style table and indexing scheme used for edit text, this data is not space-efficient; each style run gets defined by a full 22 bytes of information. However, this makes it portable between applications and is the format of a 'styl' resource, which you might come across in the desk scrap.

When you use **TEStylPaste** or **TEStylInsert** or **SetStylScrap**, **TextEdit**

THINK Reference © 1991-1992 Symantec Corporation

applies the formatting of an StScrpRec to some text. On the older **TEInsert** function and the **TEKey** function, **TextEdit** applies the style of the character to the left of the insertion point (or the first character of the selection range) to the inserted text. **TextEdit** also maintains a null style scrap, consisting mainly of a single ScrpSTElement. This style is used as the default for cases where there is no obvious style to use.

THINK Reference © 1991-1992 Symantec Corporation