

## An Overview of Justification Under QuickDraw GX

QuickDraw GX defines the process of justifying text in somewhat different terms than a typographer might be used to. Each glyph in the font is assigned various characteristics that control how that glyph is used during the process of justifying a line of text. This section defines some of the terms used in GX-related discussions of justification.

For the most part, when you specify factors in the justification table, you're specifying ems (i.e. points for a 1-point glyph). Thus, for example, if you identify a range of glyphs as permitting the addition of 0.25 on the right-hand side, that means that 40-point versions of those glyphs can add up to 10 points of whitespace (0.25 times 40). Numbers you specify for the grow case should be positive, and represent the upper limit on the extra amount of spacing (possibly whitespace, and possibly other effects) that can be added on one or both sides of a glyph or range of glyphs — the two sides of the glyph are separately controllable. Numbers specified for the shrink case should be negative, and represent the amount of spacing that can be removed.

Glyphs are assigned a justification priority that can be one of four values. The highest value is kashida priority, followed by whitespace, intercharacter, and finally the lowest priority (called the null priority). Glyphs at a higher priority have space added (or removed) up to their limits, before any glyphs at lower priorities are touched. This means that any glyphs that you wish to have participate early in the justification process should get higher priorities. Thus, because whitespace is a higher priority than intercharacter, justifying lines of text will usually only affect the whitespace glyphs on the line, leaving the intercharacter spacing alone. Only if the amount of extra space being distributed to the whitespace would exceed the specified glyphs' limits would the next lower priority level be used.

You have a lot of flexibility in assigning these priorities to glyphs. The simplest way is to always associate a glyph with a single priority and set of values. However, if you want more flexibility, a state table can be specified to give glyphs different priorities and factors in different contexts.

Many other kinds of special effects can be specified in the justification table. For example, ligature decomposition can be specified as happening automatically when a certain amount of whitespace gets added (or removed) from around a ligature. When whitespace amounts get too wide, you might want to designate glyphs as ductile (that is, they have a variant axis that changes their widths), stretchable (that is, they can be stretched mechanically), or capable of being substituted with different glyphs to make up some of the gap.

Here are some examples of the kinds of effects that you can specify via a justification table in your font:

- Ligatures that are both ductile and decomposing. Consider making your ligatures ductile so that the stems move apart under justification. Clearly there will be a point where this no longer looks good, and you'd like the ligature to break apart. You can do both of these with the same glyph.
- Substitution of wider forms. Rather than always adding whitespace, consider substituting specially designed alternative glyphs that can take up some

of that extra

space. For example, a finial ‘e’ might normally only be expected to appear at the end of the line, because the tail flourish would look bad in the middle of the line. However, if a lot of whitespace is being added to a line, consider allowing the finial form to take up some of that space.

- Cursively connected (script) Roman fonts. Traditional Arabic typography justifies text through the use of special extender bars. With GX, there’s no reason why Roman text can’t do that. Imagine a Poetica™-like font where the connections gracefully extend themselves under justification, while maintaining the connections between letters.
- Text that automatically copyfits itself. Specify a ductile axis of width for regular body text letters, so that (within limits you specify) the text automatically widens itself subtly to justify a line.
- Whitespace that behaves differently in different contexts. You might want single whitespace glyphs to be amplified differently than groups of whitespace glyphs. Alternatively, you might want whitespace that appear next to letters that already have a lot of sidebearing to be amplified differently than whitespace next to letters with long straight sides.
- Decomposition of ligatures into other forms. Since you specify the glyphs that a ligature decomposes into, these don’t have to be the same as the glyphs that originally constituted the ligature.
- Adding intermediate glyphs. You can specify the addition of a bullet, fleuron or some other decorative glyph between pairs of text glyphs that are being justified wide, for special visual effects.