

# Sample phrasing to add to layout feature descriptions

The examples below are not meant as an exhaustive set; that can be fleshed out later following the direction set forth here. But clearly some specifics are important for the discussion, so here they are. I believe all this is little more than common sense, and fairly light weight. In specification terms, these are recommendations (“should” rather than “must”). Hopefully these examples can ease some of the concerns that have been raised, and still provide useful guidance for both font developers and layout engine developers.

## **cpsp**

If this feature is implemented in a font, it should increase the advance width of all the default forms for capital letters present in the font, regardless of script. In most designs it should also cover all full-height punctuation (e.g. exclam, question, braces, brackets, and parentheses, and the ampersand). In many designs it’s also appropriate increase the advance width of any non-tabular figures intended for use with the capitals.

## **smcp**

If this feature is implemented in a font, it should provide small-cap alternate forms for all alphabetic lowercase characters present in the font, regardless of script. If the font contains figures and currency symbols designed for use with small capitals, those should also be included in the feature. In general this feature will not apply to full-height punctuation (e.g. exclam, question, braces, brackets, and parentheses, and the ampersand), because those may be needed for mixed settings of large and small capitals. Note: Some layout engines convert lowercase characters to capitals and apply ‘c2sc’ instead of using ‘smcp’. This avoids some casing issues not well-handled by straight substitution, but can introduce new issues that may not have been anticipated by the font developer.

## **c2sc**

If this feature is implemented in a font, it should provide small-cap alternate forms for all alphabetic capital characters present in the font, regardless of script. If the font contains figures and currency symbols designed for use with small capitals those should also be included in the feature. Small-cap forms of the full-height punctuation (e.g. exclam, question, braces, brackets, and parentheses, and the ampersand) are also recommended in this feature, as it will be used for settings in which all the text is set in small capitals.

## **sup**

If a font provides superior forms of alphabetic letters, this feature should substitute those forms for a-z (U+0061-U+007A) and grave (U+00E8). Suitable forms of basic punctuation (e.g. space, comma and period) are also recommended. In some cases the font may choose to supply superior forms for other letters, including those from scripts besides Latin, but layout engines may not expect these. If a font provides superior forms of figures, this feature should substitute those forms for 0-9 (U+0030-U+0039). Suitable forms of marks used for figure separators (e.g. space, comma and period) should also be provided. Superior forms of currency characters are optional.

## **frac**

If a font provides small (e.g. superior) forms of figures, this feature should provide code for assembling these into diagonal (a.k.a. vulgar) fractions for arbitrary strings of figures and figure separators.

## **sinf**

If this feature is implemented in a font, it should provide small figure variants for 0-9 (U+0030-U+0039) positioned somewhat below the font’s main baseline. Suitable forms of the figure separators (e.g. space, comma and period) may also be included.

## **unic**

If this feature is implemented in a font, it should provide variants of all alphabetic lowercase characters present in the font that would otherwise have ascenders or descenders, regardless of script. Other characters may also use alternate forms. Alternatively, the designer may elect to replace glyphs for all lowercase letters with unicast forms of a small-cap size and unicast design. This feature will generally not include alternates of punctuation characters but may include specially-designed figures if present.