L2/25-261R

Proposed updates to Unicode line breaking

Kent Karlsson

2025-10-27

The purpose with this proposal is to improve the (automatic) line breaking behaviour in some very common cases. 1) Not to have any line break opportunity in numerals that use SPACE as group separator (a common way of formatting numerals). 2) Not to break just after an (ambiguous) quote mark, nor just before an (ambiguous) quote mark, unless there is a SPACE giving a break opportunity (French typography here is an aberration but already handled; for many other languages there is currently a problem giving spurious line breaks). 3) Make some punctuation's (IS) behaviour like that of EX when they are not actually IS. 4) Allow line break after SOLIDUS in most cases, allowing line breaks in word alternates as well as for URLs/path names.

NL is equivalent to BK, replace NL by BK, and remove NL from rules

Plese **replace NL by BK** (in LineBreak.txt), since NL is the same as BK line breaking opportunity-wise. Then, since NL is no longer used, NL can be removed from all line break rules in UTS 14.

But there are still a few other specification changes that should be done, to get "good" (or at least somewhat better) automatic line breaking behaviour. There may be other changes that would be recommendable, but here we look specifically at IS and QU line break properties.

There is no break at eot

LB3 Always break at the end of text.

! eot

This should be "**Never break at the end of text**"; and "**× eot**"; which is what all text editors (plain as well as "higher level") that we have encountered actually implement.

Furthermore, consider an *empty* text "sot eot". To avoid contradiction, here: not have both "must not break" and "must break" (current rules) in this case, just have "must not break" (proposed rules).

These two rules are designed to deal with degenerate cases, so that there is at least one character on each line, and at least one line break for the whole text.

That sentence does not make any sense; please delete.

Handle Line break category IS better

A) Change to have category IS (change in LineBreak.txt)

```
002F            ; IS # Po          SOLIDUS
FF0C            ; IS # Po          FULLWIDTH COMMA
FF0E            ; IS # Po          FULLWIDTH FULL STOP
FF0F            ; IS # Po          FULLWIDTH SOLIDUS
FF1A..FF1B      ; IS # Po      [2] FULLWIDTH COLON..FULLWIDTH SEMICOLON
```

B) Add to mapping table in LB1 in UTS 14

EX	IS	If the IS character is NOT between decimal digits (of the same script) NOR between one-letter sequences, then map the line break category to EX. Could allow FULL STOP (and only FULL STOP) to **stay** IS **if** between a SPACE and a (ASCII!) decimal digit (though that is still an improper way of writing decimals). ~~Could allow SOLIDUS (and only SOLIDUS) to **stay** IS **if** between a "n" or "N" and an "a" or "A" and no letters before the "n" nor after the "a" (oddball English abbreviation).~~ *[more general rule in green background above]*
IS	SP	If a SPACE is between decimal digits (of the same script), then map its line break category to IS. (Ideally one should use NARROW NO-BREAK SPACE or PUNCTUATION SPACE for digit grouping, but that is an ideal. Most users will use ordinary SPACE.)

(With these mappings many instances of IS in the follow-on rules will be redundant and could be removed.)

These changes are intended to improve the line breaking behaviour, in particular for COMMA and FULL STOP, so that they behave like EXCLAMATION MARK (EX, which is similar to CL), i.e. as phrase/sentence ending, except when actually used between digits. It also allows for automatic line breaks within "or lists delimited by /" and path names (file names, URLs), by allowing break after / except for some cases. It also improves the line breaking for numerals when SPACE is used as group separator. (That was a simplified summary, detailed mapping above.)

Handle Line break category QU better

A) Uncalled-for QUs, 1 (change in LineBreak.txt)

These are not ambiguous quote marks to be used in running text. Change to SY.

```
275B..2760     ; SY # So     [6] HEAVY SINGLE TURNED COMMA QUOTATION MARK
        ORNAMENT..HEAVY LOW DOUBLE COMMA QUOTATION MARK ORNAMENT
1F676..1F678   ; SY # So     [3] SANS-SERIF HEAVY DOUBLE TURNED COMMA
        QUOTATION MARK ORNAMENT..SANS-SERIF HEAVY LOW DOUBLE COMMA
        QUOTATION MARK ORNAMENT
2E00..2E01     ; SY # Po     [2] RIGHT ANGLE SUBSTITUTION MARKER..RIGHT
        ANGLE DOTTED SUBSTITUTION MARKER
2E06..2E08     ; SY # Po     [3] RAISED INTERPOLATION MARKER..DOTTED
        TRANSPOSITION MARKER
2E0B           ; SY # Po         RAISED SQUARE
```

B) Uncalled-for QUs, 2 (change in LineBreak.txt and UnicodeData.txt)

These are not ambiguous quote marks. Change to OP (and Ps) and CL (and Pe) as appropriate.

```
2E02           ; QU # Pi         LEFT SUBSTITUTION BRACKET       OP&Ps
2E03           ; QU # Pf         RIGHT SUBSTITUTION BRACKET      CL&Pe
2E04           ; QU # Pi         LEFT DOTTED SUBSTITUTION BRACKETOP&Ps
2E05           ; QU # Pf         RIGHT DOTTED SUBSTITUTION BRACKECL&Pe
```

```
2E09          ; QU # Pi         LEFT TRANSPOSITION BRACKE        OP&Ps
2E0A          ; QU # Pf         RIGHT TRANSPOSITION BRACKET      CL&Pe
2E0C          ; QU # Pi         LEFT RAISED OMISSION BRACKET     OP&Ps
2E0D          ; QU # Pf         RIGHT RAISED OMISSION BRACKET    CL&Pe
2E1C          ; QU # Pi         LEFT LOW PARAPHRASE BRACKET      OP&Ps
2E1D          ; QU # Pf         RIGHT LOW PARAPHRASE BRACKET     CL&Pe
2E20          ; QU # Pi         LEFT VERTICAL BAR WITH QUILL     OP&Ps
2E21          ; QU # Pf         RIGHT VERTICAL BAR WITH QUILL    CL&Pe
```

C) Additional rules for QU:

Add these two rules (into UTS 14), with higher prio than other QU rules:

(non-SPBKCRLF) × QU

QU × (non-SP & non-SPBKCRLF)

where non-SPBKCRLF is any line break class other than SP, BK (note: NL is retired), CR, LF.

This is intended to avoid strange automatic line breaks just after (functionally) open quote mark, or just before (functionally) close quote mark. Recall that QU marks are used in different ways in different typographic traditions, including that the same character may be used both for open and close.

FULLWIDTH as a 'styling'

Notionally, (allocated) FULLWIDTH characters are "just" a style variant (annotated as <wide> in UnicodeData.txt) of the nominal characters. But this "styling" is different from all other styling found in various styling systems in that this styling also affects the line breaking behaviour. A way of formulating this yet-to-be-realized "fullwidth styling" could be:

- For Latin, Greek, Cyrillic, ASCII digits and for common (i.e., not script specific) punctuation and symbols, use CJK layout. This gives the listed characters em width (glyph centred in the em width) without changing font. This also changes line break property AL for these characters to ID (other line break properties are unchanged). Other scripts are not affected by this styling.

Not that Unicode should be pushing for this styling, but it is interesting to observe that if <wide> were actually realised as a styling, one would have to include a line break property change for the duration of the <wide> styling. And changing just AL to ID, and *leave all other line break properties unchanged* seems to be a reasonable thing to "would do".

That might we worth noting in UTS 14…