

# Requirements for Japanese text layout and XSL-FO

PRIVATE REPORT from Crossmedia publishing activity of XML Consortium  
(<http://www.xmlconsortium.org/introduce/index-e.html>)

17 April 2008

Author : members of Crossmedia publishing activity as follows

Shoji Aoki(Image YU Inc.), Takumi Ito(KYODO PRINTING CO., LTD.), Kazuhiro Kato(Image YU Inc.),  
[Kazuyuki Kuwano\(Benesse Corporation\)](#), [Hajime Sato\(MITSUBISHI PAPER MILLS LTD.\)](#),  
[Yuji Tazaki\(B.U.G., Inc.\)](#)

## 【Japanese text layout】

※We think that the more detailed requirements for almost issues would be submitted from JLTF.

Followings are the requirements which are popular in Japanese layout, and minor issues are not listed.

### line head / end wrap

- In case of text over the plural lines, there is a letter which must not be posted in the head or end of line.

The example of the line head wrap : A parenthesis to be closed, the punctuation mark.

The example of the line end wrap : A parenthesis to open.

- As well as the above, there exist original rules for every kind of the books and each publishers.

These rules should be supported.

- This issue is already implemented in XSLFormatter by Antenna's original extension.

### RUBY

- Regarding the textbooks for junior high students / a high school students, RUBY sticks to each top and bottom / left and right of the pro-character.

- This issue was already a referenced in JIS X4051 4.12.4

e.g.

くだら  
百濟  
バクチェ

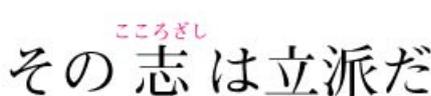
In addition, there exist many requirements for Ruby as follows.

Distance with the pro-character, A size for the pro-character, A division RUBY/ group RUBY, the position for the pro-character of the RUBY character string, Alignment when it is shorter than the width of the pro-character, Processing when an RUBY is longer than the width of the pro-character. (Having \* "CHOUTAI" processing or not, The presence of the space with a pro-character and the letter of before and after without long body processing.)

\* Modifying character width while holding character height as it is, which brings different aspect ratio of the character from the original one.

e.g.

 There is no space with a pro-character and the letter of before and after, and ruby is having "CHOUTAI" processing.

 There are spaces with a pro-character and the letter of before and after, and ruby is not having "CHOUTAI" processing.

## KENTEN

- When there is the point that should be emphasized in the sentence, characters such as "●" or "◎" are attached side by side to the character string of the pertinence with size at the same level as an RUBY.

## A spurt / hanging punctuation such as punctuation marks

- When a punctuation mark seems to come to a head of a line, it should be moved to the end of the previous line. About the placement of a moved punctuation mark, it is necessary to enables following two cases.

1. Resulted length of the line to be prescribed + 1 in length letter.
  2. Resulted length of the line including a punctuation mark should be prescribed length. (In this case, spaces between letters are clogged up at least for of the whole or a limited part)
- This issue is already implemented in XSLFormatter by Antenna's original extension.

## Letter face kerning

- This issue is already referenced in 1.2.3 <http://www.w3.org/TR/2008/WD-jlreq-20080411/> note2 c
- This feature need not implement in the body text, but required for headers.

## Setting of the space between a Japanese sentence and the European sentence

- When a Japanese letter and the European letter coexist with no spacing, it is hard to read these kind of paragraph. To avoid this difficulty, put small space between Japanese sentence and the European sentence.

- This spacing condition has a house rule.
- This issue is already implemented in XSLFormatter by Antenna's original extension.

## Adjustment of the space before or after YAKUMONO.

- YAKUMONO is the general term for special characters which represents such as a punctuation mark, a parenthesis, a centered period [ · ] and reader [...].
- Many characters in YAKUMONO has small width (the vertical typesetting, height) compared to a virtual body. Therefore, spacing between YAKUMONO and normal character looks too spaced. This issue would be more emphasized when plural YAKUMONO are placed each other.
- This issue is already implemented in XSLFormatter by Antenna's original extension with a head of a line and the letter face kerning of the YAKUMONO of the end of line.

## WARICHU processing

- Notes which are inserted in the text, the notes are narrowed size and the line feed of the letter and arrange a text for two lines to height (the vertical typesetting, width) of the degree for text one line.
- A new line can require insert a note.
- Insert a note is referred in 1.4.2 <http://www.w3.org/TR/2008/WD-jlreq-20080411/> d

e.g.

東京特許許可局 (とうきょうとつきよきよかきよくは早口)  
(言葉であり、このような組織は実在しない)

## The baseline adjustment of a European characters and the Japanese characters.

- The setting method of the baseline when a Japanese sentence European sentence coexists
- Regarding European sentence, enables to set baseline whether to bottom of text-altitude or to bottom of text-altitude + text-depth.

e.g.

Bjk } text-altitude  
      } text-depth

## Writing from left to right in writing from top to bottom.

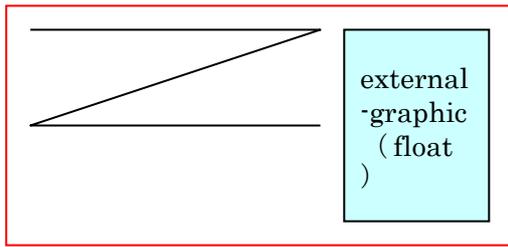
- In vertical writing text, there exist the case to setting some characters, numbers in most case, horizontally.

e.g.

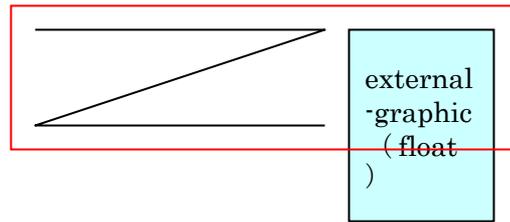
4 今  
月 日  
16 は  
日 平  
で 成  
す。 20  
年



e.g.



expected state

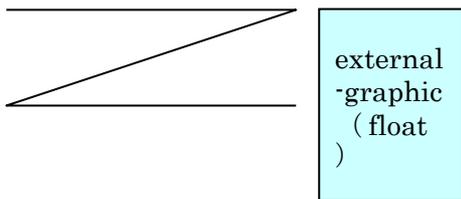


At present, the height of float object is disregarded by the border.

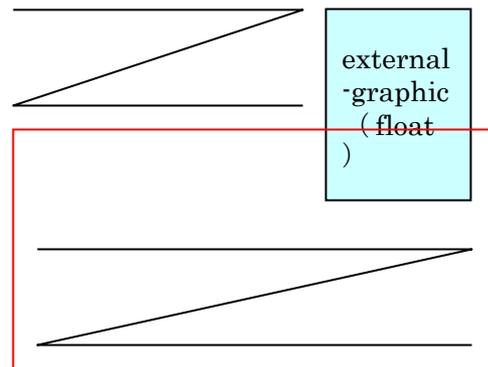
## Relation of the float object and the border 2

• When block object which contains character strings runarounds the float object which contains float object, and following block object which also contains character strings doesn't runarounds the float object and it has the border, the border is crossing external-graphic. The interval(length) from the previous block object should be given to priority.

e.g.



expected state



At present, the interval with the previous block object is given to priority.

## Adjustment of plural marginalia that sticks vertically next to each other (collision of plural marginalia)

• If there are plural marginalia (at present, we use floats as alternatives) that sticks vertically next to each other and if the distance between both of parts in character string that marginalia shows is shorter than height of the upper marginalia, we expect that the position of lower marginalia is lowered more than fixed position because not to interfere to the upper marginalia.

- Maybe supporting marginalia enables this requirement.
- There is already a reference in 2.2.3.1 <http://www.w3.org/TR/2008/WD-xslfo20-req-20080326/>
- There are lots of books for which many marginalia are used compared to the footnote like teaching materials which are used by junior high school students and high school students. It is highly recommended to implement this feature.

## Baseline grid

- Add support for “baseline grid”.
- At present, in case of horizontal composition, if one side page has table or figure or other font size's paragraph, vertical position of lines of both page of a two-page spread are not aligned each other below table etc are positioned.
- There is already a reference in 2.2.4.2 <http://www.w3.org/TR/2008/WD-xslfo20-req-20080326/>
- There is no reference in JIS X4051

## Plural columns and column spanning

- When there are more than three columns, we want to arbitrarily set that number of span.
- There is already a reference in 2.2.11.2 <http://www.w3.org/TR/2008/WD-xslfo20-req-20080326/>

## Expansion of coverage of “column-count”

- Add support for “column-count” at not only “region-body” but also “block” and “block-container”.
- There is already a reference in 2.2.11.3 <http://www.w3.org/TR/2008/WD-xslfo20-req-20080326/>
- This issue is already implemented in XSLFormatter by their original extension.

## Line positioned between Columns

- Add support for drawing line between columns.
- This issue is already implemented in XSLFormatter by their original extension.

## Enables to getting and using information which will be generated only after formatting(pagination stage).

- Add support for getting the information such as number of lines of the arbitrary paragraph, or physical size of the arbitrary external-graphic that was generated after format, and using the information for format another part.
- There is already a reference in 3 <http://www.w3.org/TR/2008/WD-xslfo20-req-20080326/>
- As a result, it comes to be able to do a detailed control such as orphans, widows, changing page with block object in the back and forth of each block object.

## Tab

- At present, if we want to realize the function equal with tab, we must use “inline-container” alternatively and set detailed properties.
- There is already a reference in 5.5 <http://www.w3.org/TR/2008/WD-xslfo20-req-20080326/> and JIS X4051

## Rounded corners

- Add support for rounded corners to rectangles such as block-container, inline-container, block, inline.

- Currently, when we want to express rectangles which has rounded corners, we made them as figures, and included as “external-graphic”.

- There is already a reference in 9.5 <http://www.w3.org/TR/2008/WD-xslfo20-req-20080326/>

- This issue is already implemented in XSLFormatter by their original extension.

## Spot Color

- Add print specific properties that specify spot color.

e.g.

“DIC xxxxx” “Pantone yyyyy”

- This requirement supplements 9.6 <http://www.w3.org/TR/2008/WD-xslfo20-req-20080326/>

This concludes our requirements.