The Asian Journal of TEX, Volume 2, No. 1, April 2008
Article revision 2008/3/16

# pTEX and Japanese Typesetting

Haruhiko Okumura 奥村 晴彦
Faculty of Education, Mie University
514-8507, Japan
okumura@edu.mie-u.ac.jp

KEYWORDS pTEX, Japanese typesetting, *jis* font metric, *js* document classes.

ABSTRACT We describe the rules of Japanese typesetting and how the combination of ASCII's pTEX, Kobayashi's *jis* font metric, and the present author's *js* document classes implements them.

## 1 Introduction

In 1987, Yasuki Saito (斉藤 康己), of the then public corporation Nippon Telegraph and Telephone (NTT), developed jTEX [1], often called NTT jTEX, an extension of TEX which could typeset Japanese text. jTEX used a subfont scheme, splitting up a Japanese character set into 33 TEX fonts, each containing at most 256 characters.

In the same year, Shunji Ohno (大野 俊治) and Ryoichi Kurasawa (倉沢 良一), of the technical publisher ASCII Corporation, developed ASCII Nihongo TEX. It was a true multibyte extension of TEX. Its extended font format allows all Japanese characters to be incorporated in one font.

Three years later, ASCII's Hisato Hamano (濱野 尚人) [2, 3] extended it to enable vertical typesetting. The new version of Nihongo TEX was named pTEX ("p" for publishing) to distinguish it clearly from jTEX. In 1995, pTEX was revised in accordance with TEX 3.0, and pLATEX2$_\varepsilon$ was developed [4].

Although the typesetting mechanisms built into pTEX was very carefully thought out, the Japanese font metric that came with pTEX was less than satisfactory, especially for Japanese punctuation and quotation marks. To circumvent this problem, in my first attempt at typesetting a whole book [5] with pTEX, I resorted to using punctuation and quotation marks from Latin (Computer Modern) fonts, and squashing Japanese glyphs vertically by 10 percent so that the Japanese and Latin glyphs have about the same height [6].

In 1993, the Japanese Industrial Standard for typesetting, JIS X 4051 [7] was published. In carefully reading the standard, I understood what were wrong with pTEX, and asked Hajime Kobayashi (小林 肇), the TEXnician who printed my book at Tokyo Shoseki Printing, to develop a completely new pTEX font metric that conformed to the standard. The result was *jis* font metric [8].

Armed with the new font metric, I started developing new *js* document classes [9, 10, 11, 12, 13], such as `jsarticle.cls` and `jsbook.cls`, on the basis of the standard

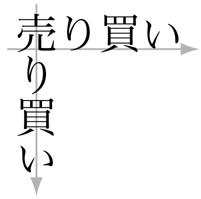


FIGURE 1. Horizontal and vertical typesetting. Note that some OpenType fonts, such as *Hiragino* used here, have different glyphs for each direction. Shuzaburo Saito's *otf* package can access these extra glyphs of OpenType fonts.

pLATEX2$_\varepsilon$ document classes, `jarticle.cls` and `jbook.cls`, which were in turn derived from LATEX2$_\varepsilon$ `article.cls` and `book.cls`. The name *js* purports to stand for "Japanese Standard."

In 2003, Shuzaburo Saito (齋藤 修三郎) developed the pLATEX2$_\varepsilon$ *otf* package [14], which consisted in a new set of virtual fonts that enable pLATEX2$_\varepsilon$ to use OpenType Japanese fonts. The two macro commands provided by the package, `\UTF{}` and `\CID{}`, outputs a character for the given 16-bit Unicode and Adobe-Japan1-5 CID (character identifier) numbers, respectively. The font metrics accompanying the package was basically *jis* font metric. Since then, several filter scripts were developed that convert UTF-8 text into traditional JIS X 0208 text with embedded `\UTF{}` (and `\CID{}`) macros for characters outside JIS X 0208. Thus, the *otf* package virtually enabled pTEX to handle Unicode text.

In 2006, Nobuyuki Tsuchimura (土村 展之) developed new pTEX implementations, ptetex [15] and ptexlive, which could handle UTF-8 inputs, although the character set remained traditional JIS X 0208. Characters outside the JIS X 0208 set were converted to the `^^ab` format, so that suitable macros could typeset the appropriate characters.

Subsequently, Takuji Tanaka (田中 琢爾) started developing upTEX [16], a true Unicode implementation of pTEX. Development of upTEX is believed to be a truly important project, but since it is still at an "alpha" stage, we shall not delve into it here.

In what follows we describe the Japanese typesetting rules and how pTEX, *jis* font metric, and *js* document classes implement them. We note that while most of the rules can be implemented with Omega and OTP [17], the ability of pTEX remains unsurpassed.

## 2 Japanese Typesetting and pTEX

Traditional Japanese text is typeset vertically (top to bottom, right to left), but most technical documents are set horizontally (left to right, top to bottom, as in English); see Figure 1.

Japanese characters consist of

- about fifty (83 including variations) phonetic characters called *hiragana*: ぁあぃいぅうぇえぉおかがきぎくぐけげこごさざしじすずせぜそぞただちぢっつづてでとどなにぬねのはばぱひびぴふぶぷへべぺほぼぽまみむめもゃやゅゆょよらりるれろゎわゐゑをん (rarely-used 84th variation is ゔ)

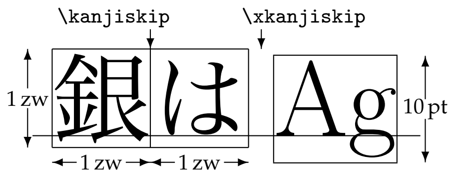


FIGURE 2. An example of mixed Japanese and Latin text. "銀は Ag" stands for "Silver is Ag".

- about fifty (84 including variations) alternative phonetic characters called *katakana*: ァアィイゥウェエォオカガキギクグケゲコゴサザシジスズセゼソゾタダチヂッツヅテデトドナニヌネノハバパヒビピフブプヘベペホボポマミムメモャヤュユョヨラリルレロヮワヰヱヲンヴ
- thousands of Chinese-origin ideographic characters called *kanji* (漢字)
- punctuation, quotation, and other marks. The comma and period are either "、" and "。" (for vertical and some horizontal text), "，" and "．" (for scholarly horizontal text), or "，" and "。" (for official horizontal documents). The quotation marks are either 「 」 for ordinary quotations, and 『 』 for quotations within quotations and often book titles. Other quotation marks such as “ ” and 《 》 may be used for various purposes.

Japanese character sets and encoding methods are provided by Japanese Industrial Standard (JIS). The basic set, JIS X 0208 (1978, revised 1983, 1990, 1997), originally called JIS C 6226, now consists of 6,879 characters. Its new superset, JIS X 0213 (2000, revised 2004), consists of 11,233 characters. They are now subsets of Unicode and ISO/IEC 10646. We note that these international standards unify similar characters used in various Asian regions into one code point (*Han* unification).

In what follows, we concentrate on horizontal typesetting, because almost all technical documents are set horizontally.

Figure 2 shows a composed text consisting of two Japanese and two Latin characters. The first Japanese character, 銀, is a *kanji*, and the second, は, a *hiragana*. These ordinary (*zenkaku* or fullwidth) characters are designed on invisible, imaginary square boxes. The width of the box of the currently selected font is defined to be 1 zw (*zenkaku* width). As can be seen in Figure 2, the baseline divides each square box typically by the 88 : 12 ratio, whereas the height-depth ratio for Computer Modern Roman, the default Latin font for both TEX and pTEX, is about 3 : 1. For Japanese and Latin characters to mingle coordinately, the height plus depth of the Latin font (i.e., 1 em) should be somewhat larger than that of the Japanese font (1 zw). The 10-point *js* document classes use 10 pt (about 3.5146 mm; 1 pt = 1/72.27 in for TEX and pTEX) Latin font with 13 Q (13 quarter-millimeter = 3.25 mm) Japanese font. The choice is partly derived from the fact that many Japanese books are typeset with 13 Q fonts. The original choice by the pTEX developers was 9.62216 pt (about 3.3818 mm) Japanese for 10 pt Latin. As a comparison, the default font size of Microsoft Word in the Japanese environment is 10.5 pt (1 pt = 1/72 in) for both Japanese and Latin characters.

Japanese text has no interword spaces. In order to break lines and justify each line on both sides, pTEX automatically inserts a glue, called `\kanjiskip`, between Japanese characters. Despite the name, `\kanjiskip` is inserted between any adjacent Japanese


FIGURE 3. Punctuations and glues.


FIGURE 4. Quotation marks and glues.

characters, *kanji* or otherwise, except when the font metric inserts a glue or kern, as explained later. The natural width of `\kanjiskip` is usually set to zero. The original pTEX setting was `0pt plus .4pt minus .5pt`, but since it is desirable that the glue cannot shrink so much as it can stretch, *js* classes set it to `0zw plus .1zw minus .01zw`.

Figure 2 shows another glue, `\xkanjiskip`, inserted automatically between Japanese and Latin characters. Traditionally it is set to 0.25 zw ± some small amount (hence it is often called *shibuaki*, or quarter-space). A reasonable choice is to equate it to the interword space for the current Latin font, because many people write either “銀は`Ag`” or “銀は␣`Ag`” interchangeably, and the results should be the same. Another choice is to equate the width of a Latin digit plus twice `\xkanjiskip` to 1 zw, because combination of Japanese + digit + Japanese, such as “第 1 回” occurs very often, and this combination looks best when the glue-digit-glue combination occupies the same width as one *kanji*. Current practices, however, tend to prefer smaller values for `\xkanjiskip`, even zero.

Japanese punctuation marks are treated differently from normal letters. Irrespective of the original width of the invisible box on which the mark is designed, the box is truncated or extended to 0.5 zw from the left edge (Figure 3, left). If the punctuation is followed by a normal letter, then *jis* font metric inserts a glue of natural width 0.5 zw, shrinkable to 0 zw (i.e., `\hskip 0.5zw minus 0.5zw`; the rectangle 3 of Figure 3). In Figure 3, `\kanjiskip` glues are inserted at points 1, 2, 4, 5, and 6, but not at 3, where a glue or kern is inserted by the font metric.

Moreover, the *kinsoku* (nobreak) rule dictates that line must not break just before punctuation marks. To ensure this, infinite (10000) penalties are inserted before punctuation marks (at points 2 and 6 of Figure 3). The *kinsoku* penalties can be controlled by the pTEX primitive `\prebreakpenalty`:

```
\prebreakpenalty`、=10000
\prebreakpenalty`。=10000
```

Another important fact must be explained for Figure 3. If this sentence comes at the end of a paragraph, then a penalty named `\jcharwidowpenalty` is inserted before the last ordinary letter (at point 5 in this case). This is necessary to prevent a “widow” line consisting of only one ordinary letter (disregarding punctuation and other marks). The default value for `\jcharwidowpenalty` is 500.

Japanese quotation marks are treated similarly to punctuation marks. As Figure 4 shows, the box for the opening (closing) quotation mark is truncated or extended to 0.5 zw from the right (left) edge. If the opening quotation mark is preceded by a normal letter, then *jis* font metric inserts a glue of natural width 0.5 zw, shrinkable to 0 zw

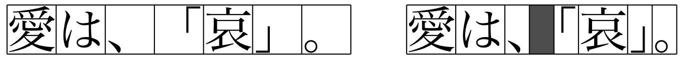


FIGURE 5. Wrong (left) and right (right) typesetting examples.

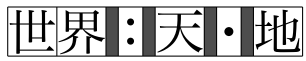


FIGURE 6. A colon and a middle point.

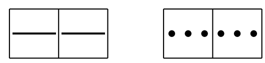

FIGURE 7. A dash and dots

(the rectangle 2 of Figure 3). Similarly, if the closing quotation mark is followed by a normal letter, then *jis* font metric inserts a similar glue. Also, in Figure 4, `\kanjiskip` glues are inserted at points 1, 3, 4, and 6, `\jcharwidowpenalty` is inserted at 5, and *kinsoku* penalties are inserted at 3 and 4 because pTEX sets:

```
\prebreakpenalty`」=10000
\postbreakpenalty`「=10000
```

Other settings with `\prebreakpenalty` and `\postbreakpenalty` can be found in `kinsoku.dtx` or `kinsoku.tex` that comes with pTEX. Some settings are overwritten by *js* document classes:

```
\prebreakpenalty`’ =10000      % was 5000
\postbreakpenalty`“=10000      % was 5000
\prebreakpenalty`” =10000      % was 5000
```

Text editors usually display Japanese text with a fixed-width font as in the left example of Figure 5, but word-processing software must typeset it as the example on the right. Microsoft Word is correct in this respect, but Apple Pages ’08 and Keynote ’08 are wrong.

Some punctuation marks, such as a colon (：) and a middle point (・) are aligned toward the center of the halfwidth (0.5 zw) box. Between these marks and normal letters are inserted quarter-width (`0.25zw minus 0.25zw`) glues, as in Figure 6.

As the final example, the double-width dash (——) and double-width dotted line (……) are composed with two fullwidth dashes and dotted lines, respectively, as in Figure 7. To ensure that the combination does not shrink, stretch, or break, *jis* font metric inserts a zero-width kern, prohibiting `\kanjiskip` to be inserted. Note that a three-point character should be designed so as to form six equidistant points when used consecutively, and similarly for a fullwidth dash, but the fullwidth dash of *Hiragino* is designed somewhat shorter than 1 zw. To compose a double-width dash in this case, one must resort to a hack like

```
―\kern-0.5zw ―\kern-0.5zw ―
```

Because there are thousands of Japanese characters, a pairwise glue/kern table is prohibitive. pTEX has an extended font metric format, often called JFM (Japanese

TABLE 1. CHARTYPEs of Japanese characters.

| CHARTYPE | width | align | examples |
|---|---|---|---|
| 0 | 1 zw | – | everything else |
| 1 | 0.5 zw | right | ‘ “ （ 〔 ［ ｛ 〈 《 「 『 【 |
| 2 | 0.5 zw | left | 、 ， ’ ” ） 】 ］ ｝ 〉 》 」 』 】 |
| 3 | 0.5 zw | center | ・ ： ； |
| 4 | 0.5 zw | left | 。 ． |
| 5 | 1 zw | – | —…‥ |
| 6 | 1 zw | – | ？ ！ |

TABLE 2. Content of *jis* font metric. "$1/2 - 1/2$" in the $(0, 1)$ cell means that a glue of `0.5zw minus 0.5zw` is inserted between consecutive characters of CHARTYPE 0 and 1 appearing in this order.

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| 0 | | $1/2 - 1/2$ | | $1/4 - 1/4$ | | | |
| 1 | | | | $1/4 - 1/4$ | | | |
| 2 | $1/2 - 1/2$ | $1/2 - 1/2$ | | $1/4 - 1/4$ | | $1/2 - 1/2$ | $1/2 - 1/2$ |
| 3 | $1/4 - 1/4$ | $1/4 - 1/4$ | $1/4 - 1/4$ | $1/2 - 1/4$ | $1/4 - 1/4$ | $1/4 - 1/4$ | $1/4 - 1/4$ |
| 4 | $1/2 - 0$ | $1/2 - 0$ | | $3/4 - 1/4$ | | $1/2 - 0$ | $1/2 - 0$ |
| 5 | | $1/2 - 1/2$ | | $1/4 - 1/4$ | | 0 (kern) | |
| 6 | $1/2 - 1/2$* | $1/2 - 1/2$ | | $1/4 - 1/4$ | | | |

* $1 - 1/2$ for vertical typesetting.

Font Metric), that groups characters into several CHARTYPEs and specifies glue/kern insertions for each pair of groups. Table 1 shows the CHARTYPEs used by *jis* and similar font metrics, and Table 2 shows the content of the metric. The classical *jis* font metric categorizes ？ and ！ as CHARTYPE 0, but the variant included in Shuzaburo Saito's *otf* package categorizes them as CHARTYPE 6 and inserts a glue (halfwidth for horizontal, fullwidth for vertical setting) if followed by normal letters. This behavior conforms to the typical typesetting rule for a question or interrogation mark at the end of a sentence; otherwise the extra glue should be suppressed by the pTEX primitive `\inhibitglue` as in "あっ！`\inhibitglue␣`と驚く" for "あっ！と驚く".

Several other pTEX primitives have yet to be explained. One is `\inhibitxspcode`. As we noted in the "銀は Ag" example, pTEX inserts a glue `\xkanjiskip` between "は" and "A". But if we modify the example to "銀は Ag。", then clearly we do not want a glue before the punctuation mark "。". Another example is "銀は「Ag」", for which we do not want glues just inside the quotation marks. To this end, pTEX has a primitive `\inhibitxspcode`. Setting it to 1 inhibits the insertion of `\xkanjiskip` on the left, 2 on the right, 3 on both sides of the specified character:

```
\inhibitxspcode`。=1
\inhibitxspcode`」=1
\inhibitxspcode`「=2
```

FIGURE 8. Comparison of old (upper) and *jis* (lower) font metrics.


\xkanjiskip can also be controlled by another primitive, \xspcode, that applies to Latin characters. In this case, 0 inhibits insertion of \xkanjiskip on both sides, 1 allows insertion on the left, 2 on the right, 3 on both sides. For example,

```
\xspcode`(=1
\xspcode`)=2
```

ensures \xkanjiskip is not inserted just inside the Latin parentheses, as in “(漢字)”.[1]

We have not provided thorough comparisons of the old and *jis* font metrics. Suffice it to give a small example, Figure 8, for a casual comparison. Note that *jis* font metric tries to conserve fullwidth-ness of Japanese characters but avoids large voids by cutting halfwidth spaces off them.

# 3 Designing document classes

## 3.1 Dimensions

In designing Japanese document classes, it is important to set every horizontal dimentions, such as \hsize and \leftmargin, to integer multiples of 1 zw. In particular, it is customary to set \parindent to 1 zw, and indent all paragraphs including the first one, although some books are typeset with zero \parindent throughout.

Note that the unit “zw” is defined with respect to the current font. If we set \parindent to 1 zw at the beginning of the document where \normalsize is in effect, and if we compose a paragraph with a \small font, then the indentation becomes greater than 1 zw of the current \small size, and the paragraph will look ugly.

To prevent this, *js* document classes redefine the \@setfontsize command so as to set \parindent, \kanjiskip, and \xkanjiskip to the appropriate values with respect to the selected font size.

Another consideration is that \baselineskip must be wider for Japanese text. Whereas plain TeX and LaTeX sets it to 12 pt, the pLaTeX2$_\varepsilon$ default document classes sets it to 15 pt (17 pt) for horizontal (vertical) typesetting, and *js* document classes (horizontal) set it to 16 pt.

Figure 9 compares typesetting results by the pLaTeX2$_\varepsilon$ default (left) and the *js* document classes. We notice that the \baselineskip is wider for *js* (which is not significant here), that the combination of two small *hiragana* “ょっ” is too tight for the default class (a well-known bug of the old font metric), and that the apparent indentation is 1.5 zw, not 1 zw, for the default class, when a halfwidth quotation mark or a parenthesis begins the paragraph.

1. In the Japanese context, Latin parentheses should not be used, because the descenders of the parentheses sticking out below the Japanese letters look ugly. Japanese parentheses should be used instead.

「どうです。すこしたべてごらんなさい」鳥捕りは、それを二つにちぎってわたしました。ジョバンニは、ちょっとたべてみて、
（なんだ、やっぱりこいつはお菓子だ。チョコレートよりも、もっとおいしいけれども、こんな雁が飛ん

「どうです。すこしたべてごらんなさい」鳥捕りは、それを二つにちぎってわたしました。ジョバンニは、ちょっとたべてみて、
（なんだ、やっぱりこいつはお菓子だ。チョコレートよりも、もっとおいしいけれども、こんな雁が飛んで

FIGURE 9. Comparison of typesetting by pLATEX2$_\varepsilon$ default (left) and *js* (right) document classes. Text: Kenji Miyazawa, *Milky Way Railroad*, ca 1927.

TABLE 3. Examples of *mincho* (`\mcfamily`) and *gothic* (`\gtfamily`), medium (`\mdseries`) and boldface (`\bfseries`) fonts.

| | `\mcfamily` | `\gtfamily` |
|---|---|---|
| `\mdseries` | 美しい印刷 | 美しい印刷 |
| `\bfseries` | **美しい印刷** | **美しい印刷** |

### 3.2 Fonts

Common Japanese fonts are classified into two families: seriffed *mincho* (明朝) and sans-serif *gothic* (ゴシック). Each family consists of lighter and heavier varieties. Thus, in the ideal world we have at least four fonts as in Table 3. This is the case when the *otf* package is used with "deluxe" option; i.e., `\usepackage[deluxe]{otf}` is specified in the preamble.

But in the real world it is often the case that we have only two fonts, a light *mincho* such as *Ryumin-Light* or *MS Mincho*, and a somewhat blacker *gothic* such as *GothicBBB-Medium* or *MS Gothic*. It is customary, therefore, to set headers and emphasized text in *gothic* and everything else in *mincho*. The default pLATEX2$_\varepsilon$ document classes and *js* classes both support only these two fonts.

If we set Japanese characters of the headers with a *gothic* (i.e., sans-serif) font, we must also use a sans-serif Latin font for the headers. For example, if we write

```
\section{MD5の脆弱性}
```

then we must get "**MD5 の脆弱性**" rather than "**MD5 の脆弱性**". This means that Japanese document classes must set headers with `\sffamily` and `\gtfamily`, and possibly `\bfseries` (although `\bfseries` has effect only when the four fonts in Table 3 are accessible). This serif/sans-serif consistency is one of the most conspicuous improvement of *js* over the default pLATEX2$_\varepsilon$ classes.

## 4 Conclusion

We tend to think that the main difficulty of typesetting Japanese text is in the large number of characters that must be handled. In fact, the number itself does not account for the difficulty; if we use subfonts and appropriate macros, we can even make TEX output Japanese characters. But making the output conform to the typesetting rules is much harder. It is hoped that this paper clarify some of the true difficulties of typesetting Japanese (and possibly other Asian languages) and help people develop a truly universal typesetting system for the future.

## References


1. Yasuki SAITO, *Report on jTEX: A Japanese TEX*, TUGboat **8** (1987), no. 2, 103-116.
2. Hisato HAMANO, *Vertical typesetting with TEX*, TUGboat **11** (1990), no. 3, 346-352.
3. アスキー出版技術部『日本語 TEX テクニカルブック I』(1990，アスキー)
4. 中野賢『日本語 LATEX2ε ブック』(1996，アスキー)
5. 奥村晴彦『C 言語による最新アルゴリズム事典』(技術評論社，1991 年)
6. 奥村晴彦『LATEX 美文書作成入門』(技術評論社，1991 年)
7. 日本工業規格 (Japanese Industrial Standard) JIS X 4051, 日本語文書の組版方法 (Formatting rules for Japanese documents), 1993, 1995, 2004.
8. 奥村晴彦『LATEX 入門——美文書作成のポイント』(技術評論社，1994 年)
9. 奥村晴彦，pLATEX2ε 新ドキュメントクラス．http://oku.edu.mie-u.ac.jp/~okumura/jsclasses/
10. 奥村晴彦『LATEX2ε 美文書作成入門』(技術評論社，1997 年)
11. 奥村晴彦『[改訂版] LATEX2ε 美文書作成入門』(技術評論社，2000 年)
12. 奥村晴彦『[改訂第 3 版] LATEX2ε 美文書作成入門』(技術評論社，2004 年)
13. 奥村晴彦『[改訂第 4 版] LATEX2ε 美文書作成入門』(2007，技術評論社)
14. 齋藤修三郎，OpenType Font 用 VF. http://psitau.at.infoseek.co.jp/otf.html
15. Nobuyuki TSUCHIMURA, *Development of a Japanese TEX Distribution 'ptetex3'*, Computer Software **24** (2007), no. 4, 40–50, in Japanese.
16. 田中琢爾，upTEX. http://homepage3.nifty.com/ttk/comp/tex/uptex.html
17. Jin-Hwan CHO and Haruhiko OKUMURA, *Typesetting CJK Languages with Omega*, TEX, XML, and Digital Typography, Lecture Notes in Computer Science, vol. 3130, Springer, 2004, pp. 139–148.