Guide to Japanese typesetting with LaTeX

Japanese TeX Development Community*

December 13, 2020

Typesetting requirements of Japanese are completely different from those of Western languages, so setting parameter values and additional hyphenation patterns are not enough.

- The `japanese` option of Babel package only defines translated replacement texts for keywords and dates (e.g. "Contents" → "目次", `\today` → "令和 2 年 12 月 13 日"); it does not care about any Japanese typesetting requirements.
- It may seem sufficient to use a package such as CJK or xeCJK, or switch to Japanese fonts with a Unicode-enabled engine; however, that's not the case.

This document describes how to get an "acceptable" Japanese typesetting result and the eminent differences between Western and Japanese typesetting requirements.

1 Short introduction: using LuaLaTeX

To obtain a properly formatted Japanese document, the easiest way is to use LuaLaTeX with LuaTeX-ja package.

- Using the jlreq class (it loads luatexja internally):

```
1  \documentclass[book]{jlreq}% loads luatexja.sty internally
2  \begin{document}
3  \chapter{日本語の文書}% Japanese document
4  こんにちは，日本。% Hello, Japan.
5  \end{document}
```

Save the above as jp1.tex and run the following command.

```
$ lualatex jp1
```

* https://texjp.org, e-mail: issue(at)texjp.org

You will get jp1.pdf which is properly formatted and a Japanese font family "HaranoAji Mincho/Gothic" is embedded.

- If you prefer classes not for Japanese, you can load luatexja directly:

```
\documentclass{book}
\usepackage{luatexja}
\begin{document}
...
```

  * Note that using non-Japanese classes will lead to unnatural page layouts for Japanese books, especially regarding line gaps and line lengths.

2 Relatively stable alternative: upLaTeX

LuaLaTeX is relatively new, and LuaTeX-ja is under active development. As a more stable alternative, you can use upLaTeX. It is a somewhat legacy implementation, which requires TFM files for typesetting Japanese characters (called JFM).

The jlreq class also supports upLaTeX, but here we provide an example using an relatively old class jsclasses, which has been widely used for over a decade:

```
\documentclass[uplatex]{jsbook}% or, \documentclass{ujbook}
\begin{document}
...
```

Save the above as jp2.tex and run the following command.

```
$ uplatex jp2
```

You will get jp2.dvi.

upLaTeX and pLaTeX (explained later) are often referred to as (u)pLaTeX. They always output DVI files. To convert DVI containing Japanese characters into PDF, the easiest way is to use dvipdfmx. If you have jp2.dvi, running the following command

```
$ dvipdfmx jp2
```

will generate jp2.pdf.

If you want to simplify the procedure, you can use the command

```
$ ptex2pdf -l -u jp2
```

which runs `uplatex` and `dvipdfmx` in sequence.

Since the default DVI driver in TeX Live is set to `dvips`,[1)] it is recommended to add a global

1) Though dvips supports DVI with Japanese fonts, additional settings for Ghostscript are required to convert the resulting PostScript file into PDF.

driver option to be passed to all driver-dependent packages (e.g. graphicx, color, hyperref):

```
\documentclass[uplatex,dvipdfmx]{jsbook}
\usepackage{graphicx}
\begin{document}
...
```

3 Using legacy pLaTeX

Some old Japanese journal classes may require pLaTeX. It is a legacy implementation which was born in 1980s, before the release of Unicode 1.0, and it only supports a limited character set, JIS X 0208 (6879 characters).

```
\documentclass{jsbook}% or, \documentclass{jbook}
\begin{document}
...
```

Save the above as jp3.tex and run the following command.

```
$ platex jp3
```

You will get jp3.dvi. Again, the DVI file can be converted into PDF using dvipdfmx.

If you want to simplify the procedure, you can use the command

```
$ ptex2pdf -l jp3
```

which runs `platex` and `dvipdfmx` in sequence.

Note for users who know the CJK package: (u)pLaTeX does *not* support the CJK package! The reasons are the followings:

- The input handling of (u)pTeX engine is different from that of Western TeX engines. Since Japanese tokens are no longer treated as active characters, the subsequent processing becomes inconsistent.
- Both the (u)pLaTeX kernel and CJK.sty redefine the command `\selectfont`, and those definitions are incompatible.

Anyway, if you use (u)pLaTeX, which provides better Japanese support, you will not need the CJK package at all.

---

4 Differences between Western and Japanese typesetting

The sections above described some practical ways to achieve "acceptable" Japanese typesetting results. In the Appendix, you will find more examples. So, what are the minimum requirements for "acceptable" results for native Japanese readers?

Listing 1　Example input using LuaLaTeX

```
%#!lualatex
\documentclass{article}
\usepackage{fontspec}
\setmainfont{HaranoAjiMincho-Regular}
\setlength{\textwidth}{250pt}
\setlength{\overfullrule}{5pt}
\begin{document}
\noindent
% first line = exactly 250pt (= 10pt x 25)
◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆\\
改行後
空白が入る。日本語が連続すると改行できず、はみ出す。\\
句読点と、（括弧）の間隔が広い。%
日本語とLatinの間が空かない。
\end{document}
```

The W3C Working Group Note "Requirements for Japanese Text Layout" [1] is one of the comprehensive documentation which describes issues of text composition in the Japanese writing system. However, they are too complicated to explain here thoroughly,[2] so we focus on the main issues regarding simple and short horizontal writing within a few lines.

4.1 Line breaks and spacing

This is the most notable aspect of Japanese typesetting, and a concept that would not be found in Western typesetting. Consider the input list, Listing 1. Using LuaLaTeX, it is allowed to include any Unicode character out-of-the-box, and they are actually displayed correctly in the output when a suitable Japanese font is declared.

However, the result (Figure 1) is not acceptable, especially for the following reasons:

- Japanese text has no interword spaces, but line break is allowed between any Japanese characters except before some punctuation marks (*kinsoku*; 禁則). The overfull rules that appeared in Figure 1 come from not knowing such line breaking rules.
- Similarly, it is common to insert a new line character after arbitrary Japanese characters in the input source. Such a new line character should not be treated as a blank space.
  Wrong: 改行後 空白が入る。 Correct: 改行後空白が入る。
- Ordinary Japanese characters are designed to have virtual full-width (*zenkaku*) bound-

2)　They include specification of basic page layout (*kihon-hanmen*; 基本版面), vertical alignment of Japanese text (*tate-gumi*; 縦組), small-sized supplementary text (*ruby*; ルビ), inline cutting note (*warichu*; 割注) etc.

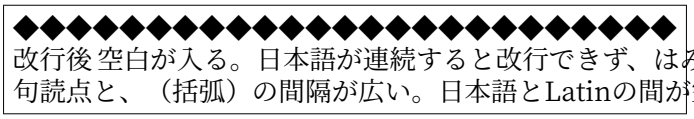

Figure 1　Output from Listing 1 (not acceptable)

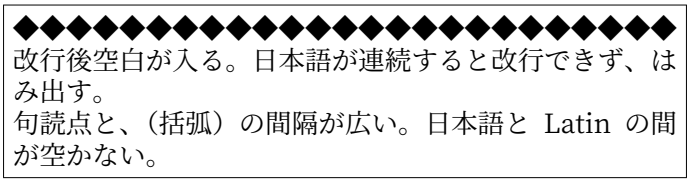

Figure 2　Acceptable output of the above text

ing box. However, spacing between adjacent punctuation marks must be truncated.
Wrong: 句読点と、（括弧）の間… Correct: 句読点と、（括弧）の間…

- It is preferred to insert a quarter em space (*shibu aki*) between Japanese and Latin characters.[3)]
Wrong: 日本語とLatinの間… Correct: 日本語と Latin の間…

Figure 2 shows an acceptable output of the same text. You will see that choosing a Japanese font on Unicode-enabled engine is not enough for obtaining properly formatted result.

All of the methods explained in the previous sections, namely

1. LuaLaTeX with LuaTeX-ja package,
2. upLaTeX,
3. pLaTeX,

are developed to support (most of) the requirements mentioned above, and their output is most likely to be acceptable. Therefore, we recommend to use one of those platforms.

4.2　Page layout (brief description)

Also, if the main language of the document is Japanese, it is common to choose a properly designed Japanese class. Unlike Latin typefaces, most of Japanese typefaces are designed in squares with a width of *zenkaku*. This difference greatly contributes to the overall page layout in Japanese typesetting.

For example, the typesetting procedure can be simplified by setting `\textwidth` to a multiple of *zenkaku*, and `\parindent` to 1 *zenkaku*. Many recent Japanese classes determine margins accordingly, and adjust other layout parameters to improve readability.

3)　Such a space is often called “xkanjiskip” glue in Japanese TeX world.

4.3 Using multi-weight typefaces for Japanese typesetting

Latin font families used in Western typesetting consist of many typefaces:

- Serif: Upright, *Italic*, **Bold**, ***BoldItalic***
- Sans Serif: Upright, *Italic*, **Bold**, ***BoldItalic***
- `Typewriter: Upright, Italic, Bold, BoldItalic`

In contrast, Japanese font families provide limited typefaces. Most Japanese font families do not provide italic or slanted shapes. Rather, these shapes look strange and have rarely been used in traditional typesetting. Also, there is no typewriter font; ordinary Japanese typefaces are inherently designed to have a fixed width.

- 明朝 (*Mincho*; Japanese Serif)
- **ゴシック** (*Gothic*; Japanese Sans Serif)

Therefore, the typeface used to emphasize in Japanese text is different from those often used in Western text:

- Normal text (in upright shape) and *Emphasized text (in italic shape)*
- 通常の文 (in *Mincho*) と，**強調された文** (in *Gothic*) か強調された文 (with *Kenten*; 圏点)

Most of common Japanese classes, including those used in the previous sections, use *Mincho* and *Gothic* with a single weight each. Nowadays, some multi-weight Japanese fonts are available for free, so using them is a good way to overcome the situation. The `[deluxe]` option of packages luatexja-preset (on LuaLaTeX) or otf (on (u)pLaTeX) supports maximum of 7 typefaces:

- 細明朝，明朝，**太明朝** (*Mincho* Light, Medium and Bold)
- ゴシック，**太ゴシック**，**極太ゴシック** (*Gothic* Medium, Bold and Extra Bold)
- 丸ゴシック (*Maru Gothic* Medium)[4)]

With the help of these packages, you can emphasize Japanese text by changing to bold, instead of changing Serif to Sans Serif.

- 通常の文 (in *Mincho*) と，**強調された文** (in *Mincho* Bold)
- 通常の文 (in *Gothic*) と，**強調された文** (in *Gothic* Bold)

4) The HaranoAji family used in this documentation does not provide *Maru Gothic* (rounded *Gothic*) typefaces.

5 Further readings

- "The LuaTeX-ja package" (luatexja-en.pdf, in English)
  - For beginners, refer to Part I "User's manual."
  - For more interested users, refer to Part II "Reference."
  - For developers, refer to Part III "Implementations."
  - Japanese edition: luatexja-ja.pdf
- "Guide to pTeX for developers unfamiliar with Japanese" (ptex-guide-en.pdf, in English)
  - For developers who aim to support pTeX/pLaTeX and its variants upTeX/upLaTeX.
- "About upLaTeX 2_ε" (uplatex-en.pdf, in English)
  - For users of upLaTeX, but a somewhat legacy document.
  - Japanese edition: uplatex-en.pdf
- "About pLaTeX 2_ε" (platex-en.pdf, in English)
  - For users of pLaTeX, but a somewhat legacy document.
  - Japanese edition: platex-en.pdf
- "Babel-Option `japanese`" (babel-japanese.pdf, in both English and Japanese)
  - Basic introduction of `japanese` option of babel.
  - Provides translated replacement texts for keywords and date.
  - Note: the translations provided by this package may not become effective on most of Western classes, due to lack of appropriate placeholders inside captions.

6 Japanese deliverables in TeX Live

There are many packages developed specially for Japanese typesetting in TeX Live; most of these are installed as part of collection-langjapanese. Only a subset of these useful packages are shown below. For more details, please refer to individual documentation.

6.1 Document classes

Japanese text can be aligned in two directions. One is horizontal direction, called *yoko-gumi* (横組), the other is vertical direction, called *tate-gumi* (縦組). The list below shows *tate* classes in green and *yoko*/*tate* classes in blue (default is *yoko*, and the class option `[tate]` switches to *tate* mode).

All the major frameworks for Japanese typesetting provide “standard classes”:

- LuaTeX-ja (luatexja by LuaTeX-ja project team)
  - Bundle: ltjsclasses (ltjsarticle, ltjsbook, ltjsreport)
  - Bundle: ltjclasses (ltjarticle, ltjbook, ltjreport, ltjtarticle, ltjtbook, ltjtreport)
- upLaTeX (uplatex by Japanese TeX Development Community)
  - Bundle: ujclasses (ujarticle, ujbook, ujreport, utarticle, utbook, utreport)*
- pLaTeX (platex by Japanese TeX Development Community)
  - Bundle: jclasses (jarticle, jbook, jreport, tarticle, tbook, treport)*

  * Due to historical reasons, ujclasses and jclasses are left with some inconvenient behaviors. You may need to adjust some layout parameters by hand, or use other classes mentioned below.

Other major classes designed for Japanese:

- Class: jlreq (by Noriyuki Abe)
  - Supports LuaLaTeX, upLaTeX and pLaTeX.
  - This class aims to implement “Requirements for Japanese Text Layout” (JLReq, 日本語組版処理の要件, [1]). The class file and necessary JFM (Japanese font metric) files are included.
- Bundle: jsclasses (by Haruhiko Okumura & Japanese TeX Development Community)
  - Class: jsarticle, jsbook, jsreport
    - Supports only upLaTeX (with option `[uplatex]`) and pLaTeX.
    - In addition to layout adjustments, these classes use newly developed JFM set (jis metrics) to avoid strange behaviors of standard pTeX JFM set (min10 etc). For more details, please refer to [2].
    - Note: when font size other than 10pt is specified, the engine primitive `\mag` is employed. Conflicts with other packages can lead to inconsistent results in graphics and/or page size. To avoid this, specify `nomag` or `nomag*` option.

6.2 Packages

- Package: plautopatch (by Hironobu Yamashita)
  - Supports upLaTeX and pLaTeX, but does no harm on other LaTeX (silently ignored).
  - (u)pLaTeX users are recommended to load this package *always*, to resolve errors and incompatibilities with Western LaTeX packages which are not aware of (u)pLaTeX.

- Package: pxchfon (by Takayuki Yato)
  - Supports upLaTeX and pLaTeX.
  - Helps users to declate physical fonts embedded in dvipdfmx's PDF output. It utilizes `\special{pdf:mapline}` or `\special{pdf:mapfile}` syntaxes.
- Package: otf (japanese-otf, japanese-otf-uptex by Shuzaburo Saito & TTK)
  - Supports only upLaTeX (with option `[uplatex]`) and pLaTeX.
  - Main feature 1: allows switching between seven different typefaces, compared to the standard of two typefaces. This function is realized by shipping seven "logical fonts" (= set of a JFM and a virtual font).
  - Main feature 2: provides commands for outputting many characters by specifying Unicode (`\UTF{....}`) or Adobe CID (`\CID{....}`), even with pLaTeX. This function is realized by mapping a huge real font from lots of smaller subset virtual fonts. A series of shorthand commands (`\aj...`) are also available.
  - LuaTeX-ja provides luatexja-otf to partially emulate feature 2 of this package.

6.3 Other support files

- ptex-fontmaps
  - Provides map files to be used with dvipdfmx and dvips.
  - To customize Japanese fonts to be used in the output, please use the command `kanji-config-updmap`.
- cjk-gs-integrate
  - Supports easy configuration of Ghostscript for CJK (Chinese, Japanese and Korean) fonts, which is necessary for processing of PostScript file output from dvips.

References

[1] W3C Working Group, "*Requirements for Japanese Text Layout*". `https://www.w3.org/TR/jlreq/?lang=en`

[2] Haruhiko Okumura, "*pTeX and Japanese Typesetting*". The Asian Journal of TeX, Volume 2, No. 1, 2008. `http://ajt.ktug.org/2008/0201okumura.pdf`

A Code Gallery — acceptable Japanese typesetting results

A.1 A simple Japanese document

Using LuaLaTeX:

```
\documentclass[book]{jlreq}% loads luatexja.sty internally
\begin{document}
\chapter{日本語がメインのシンプルな文書}
こんにちは，日本語に少しEnglishを含めてみます。
\end{document}
```

Using upLaTeX (or pLaTeX) + dvipdfmx:

```
\RequirePackage{plautopatch}% recommended
\documentclass[book,dvipdfmx]{jlreq}% driver option
\begin{document}
\chapter{日本語がメインのシンプルな文書}
こんにちは，日本語に少しEnglishを含めてみます。
\end{document}
```

Using pdfLaTeX or XeLaTeX: though they have only limited Japanese typesetting capabilities, bxjsbook (from bxjscls bundle by Takayuki Yato) aims to emulate jsclasses as much as possible. Due to technical constraints, manual adjustments are often required in the input.[5)]

```
\documentclass[pdflatex,ja=standard]{bxjsbook}
%\documentclass[xelatex,ja=standard]{bxjsbook}
\begin{document}
\chapter{日本語がメインのシンプルな文書}
こんにちは，日本語に少し~English~を含めてみます。
\end{document}
```

A.2 Changing Japanese fonts

The default Japanese font family is “HaranoAji Mincho/Gothic” which is derived from Source Han (or Noto) fonts. Suppose you want to switch to “IPAex Mincho/Gothic” ...

Using LuaLaTeX:

```
\documentclass[book]{jlreq}
\usepackage[ipaex]{luatexja-preset}% pre-defined
...
```

5) pdfTeX and XeTeX cannot insert a quarter em space (*shibu aki*) automatically, so the active character ~ is redefined to a macro to insert it manually.

Using upLaTeX (or pLaTeX) + dvipdfmx:

```
\RequirePackage{plautopatch}% recommended
\documentclass[book,dvipdfmx]{jlreq}% driver option
\usepackage[ipaex]{pxchfon}% pre-defined
...
```

A.3 More typefaces

Using LuaLaTeX:

```
\documentclass[article]{jlreq}
\usepackage[deluxe]{luatexja-preset}% multi weight
\begin{document}
\section{簡単な文章}
通常のテキストにはSerifと明朝体を使います。
\textbf{太字も使うことができます。}
\textsf{Sans Serifにはゴシック体が似合います。
\textbf{太字にも対応します。}}
\end{document}
```

Using upLaTeX (or pLaTeX) + dvipdfmx with the jlreq class: the package jlreq-deluxe is recommended.

```
\RequirePackage{plautopatch}% recommended
\documentclass[article,dvipdfmx]{jlreq}% driver option
\usepackage{jlreq-deluxe}% multi weight
\begin{document}
...
```

Traditionally, the otf package (distributed as japanese-otf on CTAN/TeX Live) has been widely used for over a decade on (u)pLaTeX.

Using upLaTeX + dvipdfmx:

```
\RequirePackage{plautopatch}% recommended
\documentclass[uplatex,dvipdfmx]{jsarticle}
\usepackage[deluxe]{otf}% multi weight
\begin{document}
...
```

Using pLaTeX + dvipdfmx:

```
\RequirePackage{plautopatch}% recommended
\documentclass[dvipdfmx]{jsarticle}
\usepackage[deluxe]{otf}% multi weight
\begin{document}
...
```

A.4 Vertical writing

As a Japanese version of lipsum, here we use bxjalipsum. It provides several sample texts.

Using LuaLaTeX:

```
\documentclass[article,tate]{jlreq}
\usepackage{bxjalipsum}
\begin{document}
\title{吾輩は猫である}% I Am A Cat
\author{夏目漱石}% Natsume Soseki
\maketitle
\jalipsum{wagahai}% 吾輩は猫である。名前... (33 paragraphs)
\end{document}
```

Using upLaTeX (or pLaTeX) + dvipdfmx:

```
\RequirePackage{plautopatch}% recommended
\documentclass[article,tate,dvipdfmx]{jlreq}% driver option
\usepackage{bxjalipsum}
\begin{document}
% The preamble of the Constitution of Japan
\jalipsum{preamble}% 日本国民は、正当に選... (4 paragraphs)
\end{document}
```

A.5 Beamer with Japanese

Note that the class beamer is not designed for Japanese.

Using LuaLaTeX:[6)]

```
\documentclass[unicode,12pt]{beamer}% ensure unicode for hyperref
\usepackage{luatexja}% for Japanese
\renewcommand{\kanjifamilydefault}{\gtdefault}% sans-serif also for Japanese
\usetheme{Copenhagen}% as you like!
\begin{document}
\section{日本語のサンプル}
\begin{frame}{日本語でBeamerを使う}
こんにちは，日本語に少しEnglishを含めてみます。
\end{frame}
\end{document}
```

Using upLaTeX + dvipdfmx:

```
\RequirePackage{plautopatch}% loads (at least) the pxjahyper package
\documentclass[dvipdfmx,12pt]{beamer}% driver option
```

6) Since hyperref 2020-08-14 v7.00f, the option unicode is enabled by default, so the explicit option is already redundant.

```
\usepackage{bxdpx-beamer}% additional settings for dvipdfmx
\renewcommand{\kanjifamilydefault}{\gtdefault}% sans-serif also for Japanese
...
```

Using pLaTeX + dvipdfmx:

```
\RequirePackage{plautopatch}% loads (at least) the pxjahyper package
\documentclass[dvipdfmx,12pt]{beamer}% driver option
\usepackage{bxdpx-beamer}% additional settings for dvipdfmx
\usepackage{minijs}% avoid pTeX's standard JFM (min10 etc.)
\renewcommand{\kanjifamilydefault}{\gtdefault}% sans-serif also for Japanese
...
```