



A Japanese Letter Class using the KOMA-Script Bundle

Typesetting Japanese with TeXLive and CJK

Gernot Hassenpflug*

National Institute of Information and Communications Technology (NICT)
4-2-1 Nukui-Kita, Koganei, Tokyo 184-8795, Japan
gernot@nict.go.jp

KEYWORDS letter, KOMA-Script, scrlltr2, CJK, Japanese typesetting.

ABSTRACT With the help of the CJK package Japanese can be typeset using a TeX distribution not specifically made for the purpose of typesetting Japanese. The KOMA-Script bundle is an alternative to the standard L^AT_EX classes and uses a European typesetting approach; the KOMA-Script classes inter-operate well with the CJK macros and the scrlltr2 letter class contains support for Japanese letter construction and envelope sizes. This article shows how these tools can be put together to write letters in Japanese, and also demonstrates an extension to the commonly available Wadalab fonts.

1 Introduction

Much development has gone into multi-lingualization and the use of CJK characters in localized TeX distributions. In Japan, the NTT Corporation's discontinued jTeX and the ASCII Corporation's current pTeX [1] are well-known. With built-in capability of JIS, SJIS and EUC-JP encodings (support for UTF-8 also exists as an add-on package) the latter supports both Japanese and English while also being actively developed. Recently, it has furthermore become possible to use CJK characters through the use of Unicode fonts; for example, with XeTeX [2]. Another approach, the topic of this article, is to use the CJK [3] package to enable special handling of CJK characters and fonts. Such an approach allows the typesetting of Japanese with, for example, the TeXLive [4] or MiK^AT_EX [5] distributions.

The KOMA-Script [6] bundle is an alternative to the standard L^AT_EX classes: it provides a wide range of functionality, is actively developed, and in addition well-documented. Use of the CJK macros is supported and the letter class, scrlltr2, caters for the Japanese writing environment and for a variety of Japanese envelopes.

An issue with typesetting Japanese with open-source and free TeX distributions is the lack of available fonts. Unless a user decides to buy fonts and accepts the problems of portability that ensue, the default fonts for use with TeXLive, for example, are

*With acknowledgements to Markus KOHM, author of the KOMA-Script bundle; and to Werner LEMBERG, author of CJK.

the Wadalab [7] fonts in Mincho, Marumoji and Gothic styles, which are usable for technical purposes but notably incomplete.

In this article I discuss first how to use the CJK package in TeXLive with the KOMA-Script `scrlltr2` letter class; second, I explain the support for Japanese letters in `scrlltr2`; and third, show an extension of the Wadalab fonts to include half-width katakana characters.

2 The KOMA-Script bundle letter class `scrlltr2`

Beautiful letters in LATEX can be typeset in a rather few number of classes thus far. The standard classes include a letter class which is quite limited in that it does not handle sections nor allow the use of floating environments. However, for quick letter writing it is sufficient and it also supports mail merge or form letters.

For more functionality and customization in letter writing, one has the following choices, available from CTAN, the Comprehensive TeX Archive Network [8]:

- `akletter` [9] for German business letters;
- `newlfm` [10] for US-centric letters, faxes and memos;
- `dinbrief` [11] for German letters;
- `brief` [12] in the Dutch `ntgclass` bundle;
- `scrlltr2` [6] in the KOMA-Script bundle.

The KOMA-Script letter class `scrlltr2` is the most advanced of these and effectively replaces the other Euro-centric classes although the `newlfm` class remains a good option for US-centric use. A further and major advantage of the `scrlltr2` class over all the other options is that it is under continuous development, so that integration with various other packages, including CJK, is guaranteed.

KOMA-Script [6], included in standard TeX distributions, offers the user European typesetting conventions as rediscovered and disseminated mainly by Jan Tschichold [13], while at the same time supporting also all the conventions of the standard classes. They act thus effectively as drop-in replacements with extended functionality and options. Development is ongoing, with an active user base and discussion forum, and detailed and extensive documentation is available in both German and English.¹

KOMA-Script is also compatible with the latest versions of important packages such as `babel` [22], `hyperref` [23], CJK and many more. Its letter class can thus take advantage of more functionality than any other existing letter class.²

2.1 `scrlltr2` letter class usage

Figure 1 gives a graphical overview of the page layout provided for by the `scrlltr2` class for the first page of a letter, with a specific letterhead and letter footer. Subsequent pages can have different headers and footers, which can be customized either using

1. It should be noted that package documentation is installed along with packages in standard TeX distributions, and can be found usually in the `$TEXMF/doc` or `$LOCALTEXMF/doc` directory tree. Alternatively, the documentation can be found at the CTAN locations of the respective packages.

2. At time of writing of this article the latest KOMA-Script release was Version 2.98.

a KOMA-Script specific page layout package, `scrpage2`, described in the KOMA-Script documentation, or a standard package such as `fancyhdr` [21].

The documentation gives numerous examples of the customization possibilities for each of the provided fields, and notes that the user can also create his or her own headers and footers from scratch via hooks provided. Such flexibility allows the user to wield tremendous power while remaining in the document space instead of tweaking the class file.

The initial setup of a letter may appear somewhat complicated and a full but basic example will be given in Section 3. In the next section are described some of the features of `scrlltr2` visible in Figure 1. For more details, and an excellent background on the typographical principles involved in the design of all the classes of the KOMA-Script package in particular, and European typography in general, the reader is strongly encouraged to read the KOMA-Script documentation and reference the website [6]. For discussions of KOMA-Script and TeX and L^AT_EX topics in general, the Usenet groups `comp.text.tex` [14] and `de.comp.text.tex` [15] are also highly recommended. Access to Usenet is available via News servers using News reader programs;³ a good free News server is Motzarella [17], but posting and access to the Usenet archives is also available via Google Groups [18].

2.1.1 `scrlltr2` features

As mentioned in the previous section, the first page header and footer of a letter can be customized with the `scrlltr2` letter class, and different headers are possible for subsequent pages. This means that a letter can take on somewhat of the form of a longer document. It is also permissible to use sections, and floating environments.

The position of elements in the letterhead can be changed; in particular, left and right elements can be switched. Provision is made for including a logo, and an example is given in the documentation.

Below the letterhead are the space for the envelope window, in which can be shown the addressee's details, and a field for supplemental data, which can be used for further detailed information about the sender or the contents of the letter. Shortened forms of the sender's particulars can be defined to be included in the return address field, as well as the full form which is usually included in the letterhead.

The `scrlltr2` class has the facility to use letter class option (.lco) files to customize the overall design of a letter: the default style, `DIN.lco`, is designed for letters on A4-size paper complying with German standard DIN 676 and suitable for window envelopes in the sizes C4, C5, C6, and C6/5 (C6 long). Several other letter class option files are provided, including for Japanese letters, described in Section 3.

Foldmarks are provided for folding a page in a way that it will fit into the envelope determined in the applicable letter class options file. Marks are provided both for horizontal folding and for vertical folding, and another mark is provided at the half-way position on the left side of the page for positioning a punch so that the letter can be easily filed. All marks can be switched on or off by the author.

The `scrlltr2` class has a facility for merging addresses from an address database to produce form letters.

3. For GNU/Linux, Gnus [16] is an excellent choice, but many web browsers can handle News connections as well.

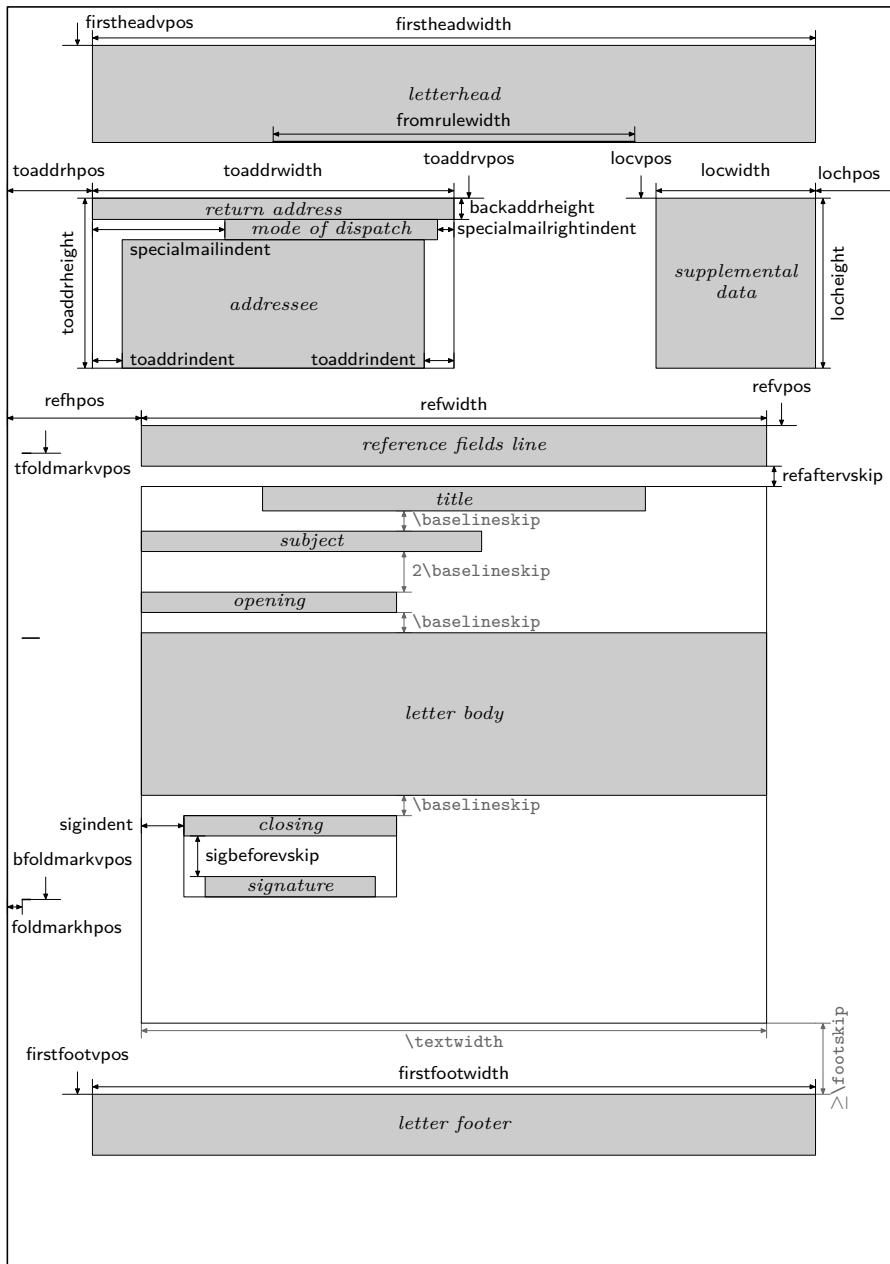


FIGURE 1. The page layout of the first page of a letter using the `scrlttr2` class shows many customizable areas, and subsequent pages can have different headers and footers. The figure is Figure 6.1 in Chapter 6, ‘The New Letter Class `scrlttr2`’, of the KOMA-Script manual at time of writing [6].

Last but not least, as previously mentioned, `scrlttr2` works together with some of the most important L^AT_EX packages for multi-lingual documents, such as `babel` and `CJK`. Through the use of the `babel` package, terminology and formats such as dates can be customized for a particular language. The same can be accomplished manually through hooks provided to the user in order that, for example, letters written in a CJK language can also use customized terminology and formats.

2.1.2 General layout of `scrlttr2` letter document

The following shows the general structure of a L^AT_EX document using the `scrlttr2` letter class. Options to the `scrlttr2` class are given to the `\documentclass` command, while further settings are given in the preamble. Thereafter, in the document but before the start of the first letter environment, settings applicable to all letters can be given, and these can be changed for individual letters in the letter environments (locally), or between letter environments (globally). The KOMA-Script documentation describes which options and settings must be defined where:

```

1  \documentclass[...]{scrlttr2}
2  ...
3  \begin{document}
4  ...
5  \begin{letter}[options]{addressee}
6  ...
7  \end{letter}
8  ...
9  \begin{letter}[options]{addressee}
10 ...
11 \end{letter}
12 ...
13 \end{document}
```

2.2 Japanese with the CJK package

The `CJK` [3] package is included in standard T_EX distributions and facilitates the use of CJK characters and fonts for a variety of Asian character encodings, as well as Unicode. Support for the free Wadalab [7] fonts is included, as are facilities for typesetting horizontally and vertically, and various text decorations such as underlining and bold face. Font changes are handled similarly to NFSS [19, 20].

In creating a document with CJK, the author needs to choose an input encoding. CJK supports a great variety, but one also needs to consider whether the fonts one wishes to use are supported in this encoding. The examples given here will all use UTF-8 encoding so that multi-lingual documents can be easily written. To support UTF-8 input encoding, font definition files for the Wadalab fonts needed to be extended slightly, and some modifications made to the vertical typesetting font definition files to position some characters better.⁴

For the case of UTF-8 encoding, one needs the following in the preamble:

4. At time of writing of this article the latest CJK release was Version 4.7.0, which supports not only the Basic Multilingual Plane (BMP) but the whole Unicode range.

```
\usepackage{CJK}
\newenvironment{Japanese}{%
  \CJKfamily{min}%
  \CJKitilde
  \CJKnospace{}}
```

This allows various CJK encoding to be used for CJK input, and defines (optionally) one language with options, with a default font family (Mincho from the Wadalab fonts), and some other options (see CJK documentation).

In the letter environment, the following notation is then used to activate the CJK processing with UTF-8 encoding at a particular point by beginning a CJK environment:

```
\begin{CJK}{UTF8}{}{}
```

and subsequently one can use any of the defined languages with:

```
\begin{Japanese}
  ...
  Japanese (and English) content
\end{Japanese}
```

Finally, the CJK environment must be ended with:

```
\end{CJK}
```

It should be noted that a language does not have to be defined: the author can simply define a family at any particular point, and similarly change other settings whenever necessary.

3 Adding Japanese letter format support to scrler2

Support for Japanese letter writing is threefold. First, the CJK macros can be used; hence, the input of CJK characters in various encoding formats is possible, and basic rules for Japanese horizontal and vertical typesetting are provided. Punctuation in vertical typesetting has been improved and in Section 4 it is shown how half-width katakana as a means of adequately writing compactly in narrow fields have been added.

Second, the **scrler2** class provides templates (the letter class option files) supporting Japanese envelope sizes and windowed envelope variants. The most common of the many window envelope variants are supported. The list of supported letter class option (.lco) files is given in Table 1. Chou (flap on short edge) 3 and 4, and You (flap on long edge) 3 and 4, are the most common Japanese envelope sizes and correspond to the European C6 size. However, window positions in Japan are different from those in European countries.

The user can define which letter class option file to use globally before all letters, or locally within an individual letter environment, as follows:

```
\LoadLetterOption{NipponLL}%
\setkomavar{myref}{NipponLL}
```

.lco file	Supported	Wsize [mm]	Wloc [mm]
NipponEL	Chou/You 3 and 4	90 × 45	122, t 12
NipponEH	Chou/You 3 and 4	90 × 55	122, t 12
NipponLL	Chou/You 3 and 4	90 × 45	125, t 12
NipponLH	Chou/You 3 and 4	90 × 55	125, t 12
NipponRL	Chou/You 3 and 4	90 × 45	198, t 28
KakuLL	Kaku A4	90 × 45	125, t 24

TABLE 1. Supported Japanese letter class option (.lco) files in the `scrlltr2` class includes variants for different window positions. *Wsize* gives the horizontal and vertical dimensions of the window, while *Wloc* gives the horizontal offset from the left (l) edge and vertical offset from the top (t) edge of the envelope.

Third, several variables have been added or modified to allow their use with non-German standards, in particular more flexible foldmarks. Another change is increased flexibility of letterhead and letterfoot customization.

3.1 Full example of Japanese letter with `scrlltr2`

The following is the complete source of an example letter output shown in Figure 2, meant only to show the general structure of a letter typeset using the `scrlltr2` letter class.

```

1 \documentclass[enlargefirstpage=on,paper=a4]{scrlltr2}
2 \usepackage[english]{babel}
3 \usepackage{CJKutf8}
4 \newenvironment{Japanese}{%
5   \CJKfamily{goth}\CJKTilde\CJKnospacing{}}
6
7 \KOMAoptions{fromalign=right, fromlogo=false, addrfield=true,
8               backaddress=off, subject=titled, subject=afteropening,
9               locfield=narrow, foldmarks=on, numericaldate=true}
10
11 \setkomavar{fromname}{ハッセンフルック・ギヤット}
12 \setkomavar{signature}{Gernot Hassenpflug}
13 \setkomavar{fromaddress}{〒181-0002 \\ 東京都三鷹市 \\
14                           犬饲2-10-1 \\ テガツの頭公園 \\ 302号}
15 \setkomavar{title}{お知らせ}
16 \setkomavar{subject}{件名：}[新しい住所]
17
18 \date{平成20年10月11日}
19
20 \begin{document}
21 \begin{CJK}{UTF8}{}{}
22 \begin{Japanese}
23
24 \begin{letter}{神戸市東灘区 \\ 西岡本7-5-1 \\ 田中文夫様}
25 \LoadLetterOption{NipponEL}% .lco file

```

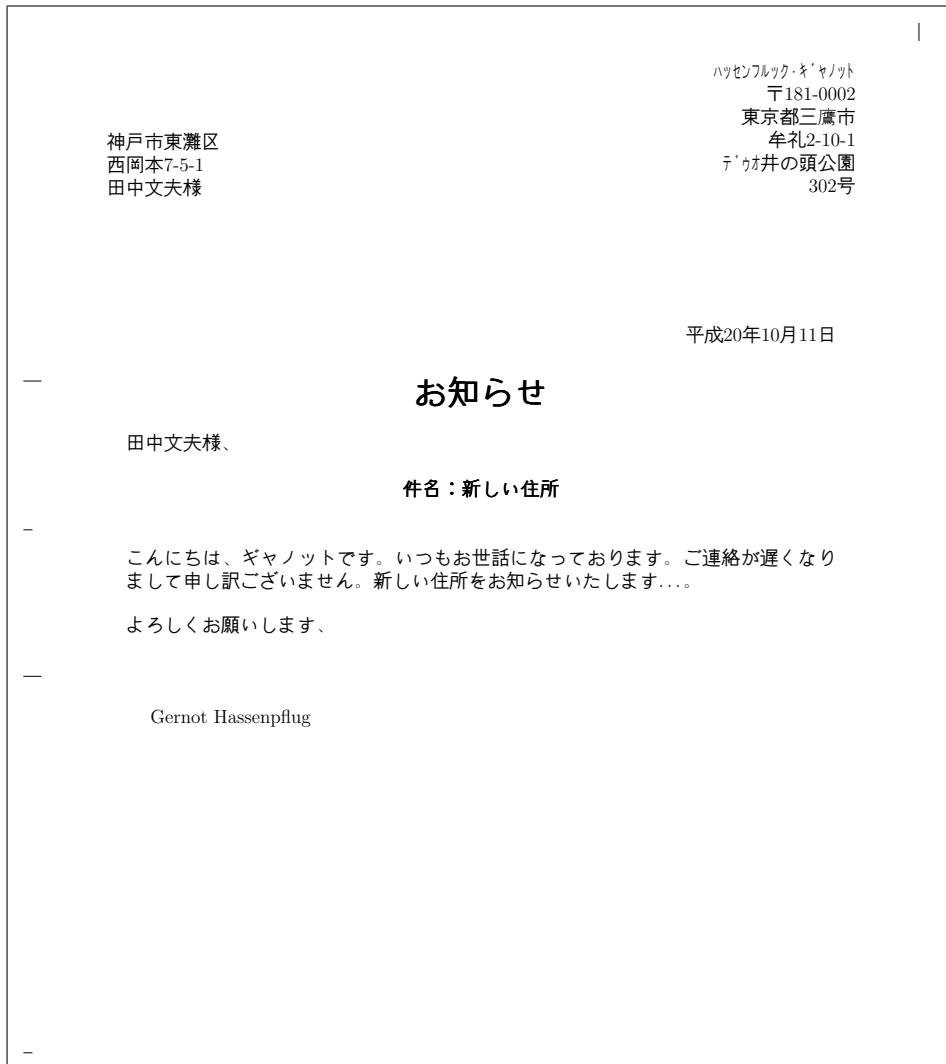


FIGURE 2. Example output from `scrlltr2` letter class is shown here demonstrating Japanese characters from the Wadalab font, and the addition of half-width katakana characters (see Section 4). Fold marks are clearly visible along the left edge and on the top right edge.

```

26 \opening{田中文夫様、}
27 こんにちは、ギヤノットです。いつもお世話になっております。ご連絡が遅く
28 なりまして申し訳ございません。新しい住所をお知らせいたします....。
29 \closing{よろしくお願ひします、 \\ \mbox{}}
30 \end{letter}
31
32 \end{Japanese}
33 \end{CJK}
34 \end{document}
```



FIGURE 3. An example of a Japanese envelope with vertical typesetting.

The addressee's name and address appear in a position where the window of the envelope specified by the letter class option file will be. The subject of the letter, below the title, is usually not written for Japanese but is included here as an example. The main points that need to be improved at present relate to the possible length of lines of the sender address, which is currently too short, and to a repositioning of the sender's name and the date. These changes can at present nevertheless be done manually by customizing the supplemental data field.

3.2 Vertical typesetting

CJK is not intended to cater completely for all Japanese typesetting needs, such as a Japanese version of TeX, for example, pTeX, does. The typesetting of specialized text such as classical Chinese literature with reading marks according to Japanese rules is not supported. A simple form of grid typesetting is possible but the inclusion of English phrases causes the method to fail. Workarounds for such issues are possible but are neither clean nor adequate.

Vertical typesetting is possible using rotation of glyphs, but this works only for fonts which have no specific character in horizontal and vertical mode. In particular, fonts which emulate brush strokes are not well-suited to such a transformation. The Wadalab fonts, however, do lend themselves well to this.

An example of vertical layout is given in Figure 3, using various boxes to contain the text and combine vertical and horizontal mode. The code of the document body is given here:

```

1 \def\redbox#1#2{%
2   \fcolorbox{red}{white}{\makebox[0pt]{\vphantom{8}\hphantom{#2}}#1\hphantom{#2}#2}}
3
4 \begin{document}
5   \sffamily
6   \begin{CJK}{UTF8}{}{%
7     \begin{Japanese}\small

```

```

8   \vspace{\baselineskip}\hspace{4.9cm}%
9   \redbox{1}{1mm}\redbox{5}{1mm}\redbox{4}{0.5mm}--\hspace{0.5mm}%
10  \redbox{8}{1mm}\redbox{5}{1mm}\redbox{0}{1mm}\redbox{4}{1mm}%
11  \vspace{0.5cm}\hspace{5.3cm}%
12  \rotatebox[origin=c]{-90}{%
13    \begin{minipage}[c]{8\baselineskip}%
14      \renewcommand{\CJKglue}{\hskip 0\baselineskip}%
15      \CJKvert\normalsize
16      東京都世田谷区 \\[-0.4\baselineskip] 世田谷
17      \makebox[0pt][l]{\hspace{0.08\baselineskip}}%
18      \rotatebox[origin=c]{90}{4}}%
19      \mbox{\phantom{--}}丁目
20      \makebox[0pt][l]{\hspace{0.08\baselineskip}}%
21      \rotatebox[origin=c]{90}{21}}%
22      \mbox{\phantom{--}}番
23      \makebox[0pt][l]{\hspace{0.08\baselineskip}}%
24      \rotatebox[origin=c]{90}{27}}%
25      \mbox{\phantom{--}}号 \\[1\baselineskip]
26      \renewcommand{\CJKglue}{\hskip 0.1\baselineskip}%
27      \Large\phantom{--}田中一郎 \phantom{--}様 \\
28      \phantom{--}田中加奈 \phantom{--}様
29    \end{minipage}}%
30  \vspace{\baselineskip}\vspace{-8\baselineskip}%
31  \vspace{1cm}\hspace{1cm}%
32  \CJKhorz {\tiny \mp181--\,0002}%
33  \vspace{0.5\baselineskip}\hspace{1cm}%
34  \rotatebox[origin=c]{-90}{%
35    \begin{minipage}[c]{5\baselineskip}%
36      \renewcommand{\CJKglue}{\hskip 0\baselineskip}%
37      \CJKvert\tiny
38      東京都三鷹市牟礼 \\ \vspace{-1mm}%
39      \makebox[0pt][l]{\hspace{0.11\baselineskip}}%
40      \rotatebox[origin=c]{90}{2}}%
41      \phantom{--}-
42      \makebox[0pt][l]{\hspace{0.11\baselineskip}}%
43      \rotatebox[origin=c]{90}{10}}%
44      \phantom{--}-
45      \makebox[0pt][l]{\hspace{0.11\baselineskip}}%
46      \rotatebox[origin=c]{90}{1}}%
47      \phantom{--} \\
48      \hspace{2.5\baselineskip} デュオ井の頭公園
49      \makebox[0pt][l]{\hspace{0.11\baselineskip}}%
50      \rotatebox[origin=c]{90}{302}}%
51      \phantom{--}号 \\
52      \hspace{1.5\baselineskip} Hassenpflug Gernot
53    \end{minipage}}%
54  \vspace{\baselineskip}
55  \end{Japanese}%
56  \end{CJK}%
57 \end{document}
```

4 Extending the Wadalab fonts

One of the shortcomings of the Wadalab fonts is the lack of half-width katakana (Unicode positions 0xFF61–0xFF9F), a customary Japanese set of characters originating in the JIS X 0201 single-byte standard and the capabilities of keyboard and screen devices of the time. As a side note, modern professional Japanese fonts often have the features of half-width, third-width and quarter-width for characters, accessible easily with such programs as Adobe InDesign.

In creating a set of such characters I used the Gothic katakana glyphs of the Wadalab fonts as a basis, and utilized the font editor program FontForge [24] to create a new font based on the Unicode BMP, copy the full-width katakana glyphs to the Unicode positions for half-width katakana, and then reduce the width of the glyphs, reposition and edit some characters, or create new ones in their place. I then output the font as a Type 1 font in compact form, with root name chosen as dgjhw. This process creates among other files a single Type 1 (.pfb) file (the total number of glyphs is less than 256), and a TeX font metric (.tfm) file.

A simple map file (`dgjhw.map`) is then created, containing solely the line

```
dgjhw dgjhw <dgjhw.pfb
```

and now the integration of the font into various encodings begins. To be used with a particular encoding it is necessary to tell L^AT_EX which glyphs to access from a given font file. With the Wadalab fonts, this is done via virtual font (.vf) files, each of which has a corresponding TFM file. In short, the VF files collect information from several fonts and when L^AT_EX requests a certain character the virtual font allows that character to be mapped to a character in one of the real fonts described in the VF. In this case, the objective is to add the information of the half-width katakana font to the VF of Wadalab Gothic, Marumoji and Mincho styles for UTF-8 encoding. Note that only one style of half-width katakana font, Gothic, is provided. Since the complete font for a particular style is divided into subfonts each holding 256 characters, the VF files I needed to modify were `udmjff.vf` (Mincho), `umrjff.vf` (Marumoji), `umcjff.vf` (Mincho), and `udgjff.vf` (Gothic).

I used the utility `vftovp` to convert each of the VF files to a human-readable form, while I used `tftopl` to convert the half-width katakana TFM to human-readable form. Then I added the font information from the latter file to each of the existing VF files and reconverted these to binary format with `vptovf`, which also created an appropriate TFM file [25]. This procedure makes it possible, after updating the TeX directories, to simply type half-width katakana characters using the UTF-8 input encoding.

Lastly, it was necessary to create, for CJK, new font definition (.fd) files for the C49 (SJIS) encoding which requests the half-width katakana font when such characters are detected in the input encoding in the user's file. The C49 encoding is also used for access of half-width katakana with the EUC-JP encoding.⁵

The half-width katakana package `hwkatakana` is available on CTAN [26], with instructions for installation; in the near future the half-width katakana characters should be incorporated directly into the Wadalab fonts and thus automatically available in a

5. This is encoding 'EUC-JP' in the CJK package, not 'JIS'.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	8															
	9															
A	。」、・ヲ	アイ	ウエ	オヤ	ユヨ	ツ										
B	-	アイ	ウエ	オカ	キク	ケコ	サシ	スセ	ソ							
C	タチツ	テト	ナニ	ヌネ	ノハ	ヒフ	ヘホ	マ								
D	ミムメモ	ヤユヨ	ラリ	ルレ	ロ	ワ	ン	・	。							
E																
F																

TABLE 2. The C49 encoding of the half-width katakana font dgjhw, located at hexadecimal positions 0xA1–0xDF. The glyphs are here displayed at a size of 12 points.

variety of standard TeX distributions. The full set of characters available is shown in Table 2.

5 Conclusion

The KOMA-Script scrlltr2 class has arguably the best support for customizable letters among all the L^AT_EX2_ε packages. CJK input is possible with the CJK package; thus, it may provide a starting point for implementing letter requirements of CJK nations.

Support for Japanese letter requirements and envelope sizes is currently still under development. Information on formal letter standards for official use at institutions, in companies or at legal firms, is scarce and no particular formats seem to be preferred. Importantly, development is needed to enable use of the powerful form letter capabilities with Japanese address formats. Similar development to support Korean and Chinese letter requirements may be possible in a similar manner.

References

1. The ASCII Corporation, *ASCII Nihongo TeX*. <http://www.ascii.co.jp/pb/ptex/>
2. Jonathan FINE, *The X₃TeX typesetting system*. <http://scripts.sil.org/xetex/>
3. Werner LEMBERG, *CJK extensions for L^AT_EX*. <http://cjk.ffi.org/> and <http://www.ctan.org/tex-archive/language/chinese/CJK/>
4. The TeX Users Group, *The TeXLive distribution*. <http://www.tug.org/texlive/>
5. Christian SCHENK, *MiK^TE_X*. <http://miktex.org/>
6. Markus KOHM, *KOMA-Script—a versatile L^AT_EX2_ε bundle*. <http://www.komascript.de/> and <http://www.ctan.org/tex-archive/macros/latex/contrib/koma-script/>
7. The Wadalab Kanji Committee, *The Wadalab fonts*. <http://www.tanaka.ecc.u-tokyo.ac.jp/~ktanaka/Font/>

8. The CTAN Team, *The Comprehensive T_EX Archive Network*. <http://www.ctan.org/>
9. Axel KIELHORN, *akletter*. <http://www.ctan.org/tex-archive/macros/latex/contrib/akletter/>
10. Paul A. THOMPSON, *newlfm*. <http://www.ctan.org/tex-archive/macros/latex/contrib/newlfm/>
11. Klaus Dieter BRAUNE and Richard GUSSMANN, *dinbrief*. <http://www.ctan.org/tex-archive/macros/latex/contrib/dinbrief/>
12. Victor EIJKHOUT, *brief*. <http://www.ctan.org/tex-archive/macros/latex/contrib/ntgclass/>
13. Jan TSCHICHOLD, *Ausgewählte Aufsätze über Fragen der Gestalt des Buches und der Typographie*, Birkhäuser Verlag, 1987.
14. Usenet group *comp.text.tex*. <nntp://comp.text.tex>
15. Usenet group *de.comp.text.tex*. <nntp://de.comp.text.tex>
16. The *Gnus News reader*. <http://www.gnus.org>
17. Wolfgang M. WEYAND, *Free News server Motzarella*. <http://www.mozarella.org>
18. Google.com, *Google Groups*. <http://groups.google.com>
19. Werner LEMBERG, *The CJK package for L_AT_EX2_ε—Multilingual support beyond babel*, TUGboat **18** (1997), no. 3, 214–224. <http://www.tug.org/TUGboat/Articles/tb18-3/cjkintro600.pdf>
20. _____, *New font tools for T_EX*, TUGboat **18** (1997), no. 3, 206–213. <http://www.tug.org/TUGboat/Articles/tb18-3/fonttool600.pdf>.
21. Piet van OOSTRUM, *fancyhdr*. <http://www.ctan.org/tex-archive/macros/latex/contrib/fancyhdr/>
22. Johannes BRAAMS, *babel*. <http://www.ctan.org/tex-archive/macros/latex/required/babel/>
23. Sebastian RAHTZ and Heiko OBERDIEK, *hyperref*. <http://www.ctan.org/tex-archive/macros/latex/contrib/hyperref/>
24. George WILLIAMS, *FontForge*. <http://fontforge.sourceforge.net/>
25. Michael DOOB and Craig PLATT, *Virtual fonts in a production environment*, TUGboat **14** (1993), no. 3, 275–281. <http://www.tug.org/TUGboat/Articles/tb14-3/tb40doob.pdf>
26. Gernot HASSENPLUG, *hwkatakana*. <http://www.ctan.org/tex-archive/fonts/hwkatakana/>