

Zpravodaj Československého sdružení uživatelů TeXu

Ken Lunde

OpenType Japanese Font Tutorial

Zpravodaj Československého sdružení uživatelů TeXu, Vol. 20 (2010), No. 3, 176–198

Persistent URL: <http://dml.cz/dmlcz/150117>

Terms of use:

© Československé sdružení uživatelů TeXu, 2010

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ*:
The Czech Digital Mathematics Library <http://dml.cz>

Abstrakt

Článek představuje klíčové technické kroky tvorby proporcionálního japonského písma nazvaného Kazuraki. Toto písmo je oproštěno od konvenčních omezení přetrvávajících v Japonsku po desetiletí. Design je inspirován kaligrafií z dvanáctého století umělce, spisovatele a jednoho z největších básníků v japonské historii Fujiwara-no-Teika. Písmo bylo poprvé oficiálně představeno 7. prosince 2009 v předváděcí místnosti firmy Adobe (North America Type Showroom).

Písmo by se mohlo stát inspirací a vzorem pro další tvůrce a písmolijny, především k vyjádření individuálního a osobitého stylu, silných výrazových prostředků a decentnějšího stylizování.

V ukázkách a přílohách tohoto článku je konkrétně použito OpenType písmo KazurakiStd-Light. Užité nástroje *makeotf*, *tx*, *mergeFonts*, *rotateFont* jsou součástí volně dostupné softwarové kolekce AFDKO (Adobe® Font Development Kit for OpenType), stáhnutelné z webových stránek <http://www.adobe.com/devnet/opentype/afdko/>.

Klíčová slova: písmo Kazuraki, proporcionální glyfy, japonské písmo, písmo OpenType, AFDKO, Adobe, *makeotf*, *tx*, *mergeFonts*, *rotateFont*.

doi: 10.5300/2010-3/176

1. Introduction

This tutorial is designed to guide Japanese font developers in building special-purpose OpenType® Japanese fonts, using KazurakiStd-Light (to be referred to as simply Kazuraki® from this point forward) as an example of how to build a genuinely and completely proportional Japanese font. The techniques, tools, and control files that are described or referenced in, or attached to, this document are tightly coupled to tools that are included in AFDKO (Adobe® Font Development Kit for OpenType) Version 2.0 or greater, which is available, at no charge, at the following URL: <http://www.adobe.com/devnet/opentype/afdko/>

Please pay attention to Section 6 on page 186 of this article, which includes information—relevant as of this writing—about compatibility with applications.

If you have any questions regarding the content of this article, please do not hesitate to contact its author, Ken Lunde (lunde@adobe.com).

© 2010 Adobe Systems Incorporated. All rights reserved. Adobe, the Adobe logo, Better by Adobe and Kazuraki is/are either [a] registered trademark[s] or a trademark[s] of Adobe Systems Incorporated in the United States and/or other countries.

2. Kazuraki Design Motivations

With the exception of the vertical-only hiragana ligatures, all glyphs in Kazuraki have corresponding horizontal and vertical forms. That is, for each character, there are two glyphs in the font. The glyphs themselves are the same, but their advances and default positioning along both X- and Y-axes are different. All glyphs needed to be replicated for vertical use, due to limitations in the ability to shift glyphs in both X- and Y-axis directions in the OpenType ‘vmtx’ table, coupled with the strong desire to make expected behavior the default—without an application depending on, or activating, any GSUB or GPOS features.

The glyph complement includes glyphs for all standard kana characters, but includes a limited set of kanji, 1,082 to be exact, suitable for creating Japanese greeting cards, menus, and other specialized uses. Kazuraki also includes a basic set of proportional Latin glyphs to aid in keyboard input. At a minimum, we recommend including glyphs that correspond to ASCII (U+0020 through U+007E). While Kazuraki is fully-functional in this limited context, it also serves as an example for building comparable fonts.

2.1. Genuine Proportional Glyphs

All horizontal/vertical glyph pairs in Kazuraki are the same, with the exception of small kana, some punctuation, and parenthetical symbols, which require different glyphs for horizontal and vertical use in conventional Japanese fonts, due to their position or orientation. Post-processing of the glyph data is used to derive the vertical versions, which are positioned along the Y-axis differently, and have different metrics.

2.2. Vertical-Only Hiragana Ligatures

Kazuraki includes a small number of vertical-only hiragana ligatures, fifty to be exact. Three are four-character ligatures, twelve are three-character ligatures, and the remaining thirty-five are two-character ones. These are activated through the use of the ‘liga’ GSUB feature. The advantage of using ‘liga’ is that most applications that perform an adequate level of typesetting also tend to automatically invoke this GSUB feature, or at least allow users to activate it through a standard UI. This allows vertical-only hiragana ligatures to be used by default in many applications.

2.3. CID- Versus Name-Keyed Structure

Kazuraki was built as a CID-keyed OpenType font. Its ‘CFF’ table is built from a CIDFont resource. Although Kazuraki could have been built as a name-keyed font, CID-keyed fonts have advantages for Japanese fonts. The primary

advantage is that a CID-keyed structure supports multiple hint dictionaries. Each hint dictionary ideally covers glyphs for specific glyph classes, and each hint dictionary can have its own hinting parameters. Using multiple hint dictionaries thus offers significant rendering advantages.

3. OpenType Table Settings & Overrides

Many of the OpenType tables require special settings for Kazuraki. This section describes the special settings, one table at a time, along with information that demonstrates how the table overrides are specified in the “features” file as used as input for AFDKO’s *makeotf* tool.

3.1. BASE

There is no special treatment necessary for the ‘BASE’ table, other than the usual ICF (Ideographic Character Face) and baseline values that should be normally specified. It is very important to include a ‘BASE’ table in all OpenType fonts. The following is the ‘BASE’ table override in the “features” file of Kazuraki:

OpenType fonts. The following is the ‘BASE’ table override in the “features” file of Kazuraki:

```
table BASE {
  HorizAxis.BaseTagList          icfb icft ideo romn;
  HorizAxis.BaseScriptList DFLT ideo -117 877 -120 0,
                               hani ideo -117 877 -120 0,
                               kana ideo -117 877 -120 0,
                               latn ideo -117 877 -120 0;
  VertAxis.BaseTagList          icfb icft ideo romn;
  VertAxis.BaseScriptList DFLT ideo 3 997 0 120,
                               hani ideo 3 997 0 120,
                               kana ideo 3 997 0 120,
                               latn ideo 3 997 0 120;
} BASE;
```

Note that only the ‘DFLT’, ‘hani’, ‘kana’, and ‘latn’ scripts are declared in the ‘BASE’ table override, based on the glyph complement of Kazuraki.

3.2. CFF

The original source data for Kazuraki is a name-keyed OpenType font containing exactly 1,450 glyphs, each with 1000-unit set widths. The same source font also contains ‘palt’ and ‘vpal’ GPOS features that specify the desired glyph metrics (horizontal and vertical set widths, and X- and Y-axis shifting values,

respectively), and these GPOS features are used as the source of the default metrics for the horizontal and vertical glyphs in the final font, which is an Adobe-Identity-0 CID-keyed OpenType font containing 2,973 glyphs (CIDs 0 through 2972).

Three AFDKO tools are used to process the original set of 1,450 glyphs: *tx*, *mergeFonts*, and *rotateFont*.

The first tool, *tx*, simply extracts the ‘CFF’ table from the source OpenType font, and converts it into a name-keyed Type 1 font, using the following command line:

```
% tx -t1 Kazuraki_2.5.otf > font.pfa
```

The second tool, *mergeFonts*, is used to convert the glyph names into CIDs, and to simultaneously synthesize the vertical glyphs from the original horizontal versions, using the following command line:

```
% mergeFonts -cid cidfontinfo cidfont.raw  
h.map font.pfa v.map font.pfa
```

The end result is an Adobe-Identity-0 CIDFont resource, named “cidfont.raw,” containing 2,673 glyphs in a single hint dictionary.

The third tool, *rotateFont*, serves to set the widths for the horizontal glyphs, and to also shift them along the X-axis. The set width and X-axis shifting values are in the ‘palt’ GPOS feature of the source name-keyed OpenType font. The following command line is used:

```
% rotateFont -t1 -rtf shift.map cidfont.raw cidfont-prop.raw
```

The end result is an Adobe-Identity-0 CIDFont resource, named “cidfont-prop.raw,” containing 2,673 glyphs, the horizontal versions of which now have proportional metrics. 300 proportional Latin glyphs are added to bring the glyph complement up to 2,973 glyphs.

As an example, let us explore the treatment of the horizontal glyph for the hiragana character “shi” (し). The glyph in the original name-keyed font is named “CID864” (named after CID+864 of the Adobe-Japan1-*x* character collection). Its ‘palt’ GPOS feature settings were as follows in the source name-keyed OpenType font:

```
position \CID864 <-223 0 -485 0>;
```

After processing by the *mergeFonts* tool, this (horizontal) glyph becomes CID+224. The ‘palt’ data shown above is used to generate the following *rotateFont* directive for the “shift.map” mapping file:

```
224 224 515 -223 0
```

The calculation is simple: the value “-223” is used as-is as the X-axis shifting value, and the default set-width value of 1000 becomes 515 after the value “-485” is added to it (a subtraction operation).

The set widths and Y-axis shifting for the vertical glyphs are specified in a ‘vmtx’ table override definition that is inserted into the “features” file. The

handling of vertical glyphs, in terms of specifying their set widths and Y-axis positions, is covered later in this document.

Once these tools have been run, and the glyphs are assigned to CIDs, and the horizontal glyphs have been set to their default set widths and X-axis positions (that is, made proportional), the CIDFont is then hinted as usual. The process of hinting also involves creating multiple hint dictionaries, ideally only one for each glyph class.

Note: The process of establishing multiple hint dictionaries in a CIDFont requires files and tools that are not included in AFDKO, and their description is intentionally (and appropriately) omitted from this article. However, mergeFonts techniques described in Adobe Tech Note #5900 (“AFDKO Version 2.0 Tutorial: mergeFonts, rotateFont & autohint”), which is among the documentation bundled with AFDKO, can be used to establish multiple hint dictionaries. Multiple mergeFonts mapping files, in which the first line each file names a hint dictionary, is the appropriate technique. And, multiple mergeFonts mapping files can serve to specify glyphs for single hint dictionaries. In fact, the proprietary tool that was used to establish multiple hint dictionaries for Kazuraki uses mergeFonts to perform this task.

The Special-Purpose Adobe-Identity-0 Character Collection

Because the glyph complement of Kazuraki does not adhere to the Adobe-Japan1-6 character collection, and because it makes little sense to extend Adobe-Japan1-6 to accommodate such special-purpose fonts, the special-purpose Adobe-Identity-0 character collection is advertised in the CFF. Although “Adobe-Identity-0” does not explicitly specify that Kazuraki is a Japanese font, other table settings, along with proven heuristics, are used to make clear the fact that it is a Japanese font.

In essence, the advantage of using the Adobe-Identity-0 character collection is that there are no preconceived notions of language or script, making it possible to build CIDFonts based on dynamic glyph sets, much like TrueType and name-keyed OpenType fonts. The technique of using the Adobe-Identity-0 character collection should not be used to build general-purpose OpenType Japanese fonts. The Adobe-Japan1-*x* character collection should be used instead.

File Size Issues

Because the horizontal/vertical glyph pairs are identical, in terms of their outlines, the subroutinization ability of AFDKO’s *makeotf* tool makes the resulting CFF table only slightly larger than that of the original source data, which contained roughly half the number of glyphs. The subroutinized ‘CFF’ table thus became approximately fifty percent the size of the unsubroutinized version.

Hinting Issues

Hinting, in terms of stem widths, is applied as usual for Kazuraki. Alignment zones, however, are another matter. The hint dictionaries for non-Latin glyph classes, such as kana and kanji, typically use the following `/BlueValues` array:
`/BlueValues [-250 -250 1100 1100] def`

However, due to the larger (taller) than usual bounding boxes of the vertical-only hiragana ligatures, the hint dictionary for the kana glyphs require different values, in order to ensure that there are no alignment zones in contact with its glyphs. The “Kana” hint dictionary of Kazuraki uses the following `/BlueValues` array:

```
/BlueValues [-1250 -1250 2000 2000] def
```

Furthermore, the “Dingbats” and “Kanji” hint dictionaries of Kazuraki uses the same `/BlueValues` array, due to the extent to which the shapes of their glyphs extend above and below the 1000×1000 em-box.

The following is the `/FontBBox` for Kazuraki:

```
/FontBBox {-326 -1179 1573 1939} def
```

It was thus critical to select `/BlueValues` values less than `-1179` (the Y-axis low point) and greater than `1939` (the Y-axis high point), especially for the “Kana” hint dictionary.

3.3. GPOS

The only GPOS features that are included in Kazuraki are ‘kern’ and ‘vkern’, for horizontal and vertical kerning, respectively. The ‘palt’ and ‘vpal’ GPOS features in the original source data served to drive the production process, to specify the horizontal/vertical set widths and X- and Y-axis shifting values. These GPOS features are not in the final form of the font, because they are not necessary. Their values were used to define the default glyph metrics.

3.4. GSUB

Kazuraki contains only four GSUB features: ‘fwd’, ‘vert’, ‘vrt2’, and ‘liga’. Although the conventional ordering of these features is ‘liga’ followed by ‘vert’ and ‘vrt2’, this font’s vertical-only hiragana ligatures necessitates a different ordering, specifically ‘vert’ and ‘vrt2’ followed by ‘liga’. As a general rule, the ordering of GPOS and GSUB features in the “features” file is important, because the same ordering is reflected in the ‘GPOS’ and ‘GSUB’ tables that are generated by AFDKO’s *makeotf* tool.

The ‘vert’ GSUB feature substitutes the horizontal forms with their vertical versions. This feature covers the majority of the font. Once the ‘vert’ GSUB feature has been applied, the vertical-only hiragana ligatures can then be applied via the ‘liga’ GSUB feature.

OpenType-savvy applications that support vertical writing automatically invoke the ‘vert’ (or ‘vrt2’, if present) GSUB feature. These same applications also invoke the ‘liga’ GSUB feature by default, which then serves to activate (or make default) the vertical-only hiragana ligatures.

3.5. OS/2

Because the special-purpose Adobe-Identity-0 character collection is used for Kazuraki, several ‘OS/2’ table fields must be more carefully specified, such as the OS/2.unicodeRange and OS/2.codePageRange fields. For Kazuraki, these settings are specified in the “features” file as the following ‘OS/2’ table overrides:

```
XHeight 423;  
CapHeight 645;  
UnicodeRange 0 1 2 5 31 33 35 36 38 48 49 50 59 62 65 68;  
CodePageRange 1252 932;
```

Note that the “XHeight” and “CapHeight” values are set to values that correspond to the proportional Latin glyphs that are in its glyph complement.

The “UnicodeRange” values correspond as follows:

0	Basic Latin
1	Latin-1 Supplement
2	Latin Extended-A
5	Spacing Modifier Letters
31	General Punctuation
33	Currency Symbols
35	Letterlike Symbols
36	Number Forms
38	Mathematical Operators
48	CJK Symbols And Punctuation
49	Hiragana
50	Katakana
59	CJK Unified Ideographs
62	Alphabetic Presentation Forms
65	Vertical Forms
68	Halfwidth And Fullwidth Forms

The “CodePageRange” value of 1252 corresponds to “Latin 1,” and 932 corresponds to “JIS/Japan.” These ‘OS/2’ table settings help to explicitly identify Kazuraki as a Japanese font.

3.6. VORG

The ‘VORG’ table is automatically generated when using AFDKO’s *makeotf* tool, and is derived from the settings and overrides of the ‘vmtx’ table. See the section for the ‘vmtx’ table for more information on ‘vmtx’ table settings and overrides.

3.7. cmap

The ‘cmap’ table for CID-keyed OpenType fonts is built using one or more CMap resources. For Kazuraki, because it is based on the special-purpose Adobe-Identity-0 character collection, special-purpose CMap resources are necessary. Because the vertical glyphs are accessible through the ‘vert’ and ‘vrt2’ GSUB features, only the horizontal glyphs are mapped from Unicode code points.

3.8. name

The ‘name’ table is built as usual, setting English and Japanese strings, as appropriate. The only exception is the name.ID=20 string, which is not necessary due to the special-purpose nature of Kazuraki. The specification of ‘name’ table strings is performed in the “FontMenuNameDB” and “features” files. Care must be taken to explicitly set Japanese as the script and language, for as many of the strings as possible, as appropriate.

For more information about specifying ‘name’ table strings for OpenType Japanese fonts, please refer to Adobe Tech Note #5149 (“OpenType-CID/CFF CJK Fonts: ‘name’ Table Tutorial”), available at the following URL:

http://www.adobe.com/devnet/font/pdfs/5149.OTFname_Tutorial.pdf

3.9. vmtx

The ‘vmtx’ table plays an absolutely crucial role in building fonts such as Kazuraki, because it is in this table that the vertical set widths are specified, along with any Y-axis shifting. Anything specified in the ‘vmtx’ table becomes default behavior. Thus, OpenType-savvy applications that support vertical writing can use such fonts without modification.

Kazuraki’s “features” file contains a very large number of “VertAdvanceY” and “VertOriginY” statements in its ‘vmtx’ table overrides. Nearly every vertical glyph required treatment by one or both of these ‘vmtx’ overrides.

As an example, let us explore the treatment of the vertical glyph for the hiragana character “shi” (し). The glyph in the original name-keyed font is named “CID864” (named after CID+864 of the Adobe-Japan1-*x* character collection). Its ‘vpal’ GPOS feature settings were as follows:

```
position \CID864 <0 -26 0 331>;
```

After processing by *mergeFonts*, this (vertical) glyph became CID+1682. The ‘vpal’ data shown above was used to generate the following ‘vmtx’ table overrides for the “features” file:

```
VertOriginY \1682 906;  
VertAdvanceY \1682 1331;
```

The calculation is simple: the value “-26” is subtracted from 880 (a fixed value that represents the top of the em-box) to become 906, which is the new origin, and the value “331” is added to the default 1000-unit width to become 1331.

4. Special Tools

A single special-purpose tool was written, in Perl, to generate all of the control files and data in a single execution. The mapping files that controlled the execution of the *mergeFonts* and *rotateFont* tools were generated by this tool, as was the “features” files containing ‘vmtx’ table overrides and all GSUB and GPOS feature definitions. The raw data to build the Unicode (UTF-32) CMap resources was also generated by this tool. Due to the large number of glyphs, and the complex relationships between them, it was important to create a tool to do this work, because doing so by hand would have been tedious, and also prone to error.

When writing a comparable tool, I found that it was very useful to maintain a mapping from the glyph names in the source font to the final CIDs. This made generating the raw data for the CMap resource a much easier task. It also made other tasks easier.

5. OpenType Control Files & Data

Once the name- to CID-keyed conversion is complete, the usual control files and data, required by AFDKO’s *makeotf* tool, must be generated or supplied. These control files and data are detailed in the following sections.

5.1. CIDFont Resource

Kazuraki’s CIDFont resource is constructed as usual, with an appropriate number of hint dictionaries, ideally one for each glyph class, and with Adobe-Identity-0 as its advertised ROS (*/Registry*, */Ordering*, and */Supplement*, which are the three entries of the */CIDSystemInfo* dictionary). As stated earlier in this document, Kazuraki’s CIDFont resource contains 2,973 glyphs, specifically CIDs 0 through 2972. Kazuraki contains exactly six hint dictionaries, named as follows, and with the number of glyphs in each in parentheses:

- KazurakiStd-Light-Dingbats (102 glyphs)
- KazurakiStd-Light-Generic (one glyph)
- KazurakiStd-Light-Kana (406 glyphs)
- KazurakiStd-Light-Kanji (2,164 glyphs)
- KazurakiStd-Light-Proportional (150 glyphs)
- KazurakiStd-Light-ProportionalRot (150 glyphs)

5.2. The “features” File

The “features” file plays an important role, in that overrides to specific tables can be made, and GPOS and GSUB features can be defined. Kazuraki contains two GPOS features, ‘kern’ and ‘vkern’, to specify horizontal and vertical kerning pairs, respectively. Four GSUB features—‘fwd’, ‘vert’, ‘vrt2’, and ‘liga’—are also included, whose relative order is important, as described earlier in this document. Lastly, the ‘vmtx’ table overrides, which are also used to build the ‘VORG’ table, serve to specify the default vertical metrics.

5.3. The “FontMenuNameDB” File

The English and Japanese menu names that are recorded in the ‘name’ table of an OpenType font are specified in the “FontMenuNameDB” file. Kazuraki’s “FontMenuNameDB” entry is shown below:

```
[KazurakiStd-Light]
f=3,1,0x411,\304b\3065\3089\304d Std
s=3,1,0x411,L
c=3,1,0x411,\304b\3065\3089\304d Std L
f=1,1,11,\82\xa9\82\xc3\82\xe7\82\ab Std
s=1,1,11,L
c=1,1,11,\82\xa9\82\xc3\82\xe7\82\ab Std L
f=Kazuraki Std
s=L
c=Kazuraki Std L
```

It is important to stress that English menu names must be set—in addition to the obvious Japanese menu names—in case such fonts are used in applications whose heuristics may cause a failure to properly use the Japanese menu names.

5.4. CMap Resources

For special-purpose fonts such as Kazuraki, only a Unicode CMap resource is necessary. Even if there are no mappings outside the BMP, a UTF-32 CMap resource is still recommended as input to AFDKO’s *makeotf* tool. For Kazuraki, the Unicode CMap resource was named “UniKazurakiStd-UTF32-H” to make

it tightly coupled with the font. This CMap resource is used solely as input to AFDKO’s *makeotf* tool, to build the Unicode ‘cmap’ subtables of the resulting OpenType font.

For more information about building CMap resources, please refer to Adobe Tech Note #5099 (“Building CMap Files for CID-Keyed Fonts”), available at the following URL:

<http://www.adobe.com/devnet/font/pdfs/5099.CMapFiles.pdf>

6. Testing & Compatibility Considerations

Kazuraki works as expected in Adobe InDesign® CS2 and greater. The horizontal and vertical metrics are respected, and proper vertical layout is supported, including the vertical-only hiragana ligatures.

Kazuraki functions in Adobe Illustrator® CS2 and Adobe Photoshop® CS2 with some limitations, specifically that the vertical-only hiragana ligatures do not function, even if the ‘liga’ GSUB feature is turned on.

In addition, these and other applications may not display Kazuraki’s name in Japanese in their font menus. Kazuraki’s name may instead display in English, using the English-language menu name strings that are specified in the ‘name’ table. Kazuraki works very well with CS3 and CS4 applications, specifically InDesign, Illustrator, and Photoshop. In fact, we recommend that CS3 and later applications be used for Kazuraki and comparable fonts.

Due to its unique (and limited) glyph complement, Kazuraki is not recommended for use as a component in these applications’ Composite Font functionality.

When developing special-purpose OpenType Japanese fonts, it is prudent to rigorously test the font with a variety of OSes and applications, to include entire document authoring workflows.

7. Glyph Synopsis

The following eleven pages provide a complete glyph synopsis for the 2,973 glyphs of Kazuraki, arranged by CID. The following list provides information about specific CID ranges:

0001–0150	Horizontal glyphs—proportional Latin
0151–1462	Horizontal glyphs—Japanese
1463–1612	Pre-rotated forms of CIDs 1–150
1613–2920	Vertical forms of CIDs 151–1462
2921–2972	Vertical-only hiragana ligatures and pre-composed double kana iteration marks

8. The Latest News

Licenses to use Kazuraki became available for sale for \$35 on Adobe's North America Type Showroom on December 7, 2009, and on their Japan, France, and Germany Type Showrooms on December 21, 2009.

Summary: OpenType Japanese Font Tutorial: Kazuraki

Adobe System's Type Engineering & Design team in Japan has developed a ground-breaking and innovative new typeface design that breaks the mold that has constrained Japanese typefaces for decades. The typeface design, created by Adobe's own Ryoko Nishizuka, was inspired by the calligraphy of the 12th century Japanese calligrapher and writer Fujiwara-no-Teika, and its final production to produce a functional OpenType font leveraged three powerful AFDKO (Adobe Font Development Kit for OpenType) tools, *tx*, *mergeFonts*, and *rotateFont*, to implement its complex metrics.

Kazuraki is unique among other mainstream Japanese typefaces in that it is fully proportional, in both writing directions. Some glyphs are wider than they are tall, and some are taller than they are wide, and this is reflected in their metrics. For this reason, and because subtle shifting is required for correct positioning of each glyph, there are separate glyphs for both writing directions. In other words, for the 1,082 kanji that are supported in the current version, the font contains 1,082 glyphs for horizontal use, and 1,082 glyphs for vertical. In addition, Kazuraki also includes a significant number of two-, three-, and four-character hiragana ligatures for vertical use.

The tutorial that is reprinted here in its entirety is designed to guide font developers in building special-purpose OpenType fonts, using Kazuraki as an example of how to build a fully-proportional Japanese font. The current version can always be accessible here:

http://www.adobe.com/devnet/font/pdfs/5901.Kazuraki_Tutorial.pdf

The Kazuraki specimen book, which demonstrates how this font can be used, is available here:

http://store4.adobe.com/type/browser/pdfs/Kazuraki_SPN.pdf

Keywords: Kazuraki font, Proportional glyphs, Japanese font, OpenType font, AFDKO, Adobe, *makeotf*, *tx*, *mergeFonts*, *rotateFont*.

*Dr. Ken Lunde (lunde@adobe.com)
Senior Computer Scientist, CJKV Type Development
Adobe Systems Incorporated, 345 Park Avenue
San Jose, CA 95120-2704 USA*

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	☒	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	0	1	2		
20	3	4	5	6	7	8	9	:	;	<	=	>	?	@	A	B	C	D	E	F
40	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
60	[\]	^	_	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n		
80	o	p	q	r	s	t	u	v	w	x	y	z	{		}	~	...	%	'	'
100	“	”	-	—	•		˘	-	;	;	¥	¤	€	¢	£	\$	¢	©	°	
120	ˆ	½	⅓	⅔	¼	¾	⅛	⅜	⅝	⅞	¬	±	×	÷	-	Ä	Ë	Ï	Ö	Ü
140	ā	ē	ī	ō	ū	fi	fl	ff	ffi	ffl	//	,	。	,	.	•	:	;	、	
160	ゝ	、	ゞ	々	ノ	○	一	—	/	~		'	'	“	”	˘	、	ゝ
180	ゝ	()	[]	[]	{ }	<	>	《 》	「 」	『 』	【 】	°								
200	'	"	あ	あ	い	い	う	う	え	え	お	お	か	か	さ	さ	く	く	け	げ
220	こ	こ	さ	さ			す	す	せ	せ	そ	そ	た	た	だ	だ	ら	ら	っ	っ
240	て	と	と	な	に	ぬ	ぬ	の	は	は	ば	ば	ひ	ひ	ひ	よ	よ	お	お	へ
260	ほ	ほ	ほ	ま	み	む	め	も	や	や	ゆ	ゆ	よ	よ	り	り	る	る	わ	わ

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

280 わ め 魚 を し づ か け ア ア イ イ ウ ウ エ エ オ オ カ ガ

300 キ キ ク グ ケ ゲ コ ゴ サ ザ シ ジ ス ス セ セ ソ ズ タ グ

320 チ チ ッ ツ ヅ テ テ ト ド ナ ニ ヌ ネ ノ ハ バ パ ヒ ビ ピ

340 フ ブ プ ヘ ベ ペ ホ ボ ポ マ ミ ム メ モ ヤ ヤ ュ ユ ヨ ヱ

360 シ リ ル レ ロ ッ ヴ ヱ エ ヲ ソ ヱ カ ケ グ ギ エ ギ 哀 愛

380 換 葵 梶 鯨 庄 扱 鮎 栗 安 暗 案 杏 似 伴 位 依 困 意 異 移

400 胃 衣 井 育 郁 磯 一 壹 稻 芋 允 員 引 飲 蔭 院 隱 右 守 鳥

420 羽 兩 卵 丑 鰻 浦 瓜 宮 管 歎 榮 永 英 液 趣 榎 内 園 宴 炎

440 煙 陸 塩 於 央 奧 押 横 王 翁 黃 尾 栢 牡 乙 温 穩 音 下 化

460 何 佃 加 可 夏 家 暇 果 歌 河 火 花 茄 荷 華 菓 蝦 過 霞 我

480 茅 賀 介 合 解 回 塊 壤 快 悔 懷 改 海 灰 界 皆 絵 芥 蟹 用

500 階 貝 外 害 街 垣 柿 谷 拵 格 殻 確 角 学 樂 掛 望 鰥 割 活

520 滑 且 鯉 株 蒲 鴨 粥 冠 寒 卷 完 官 干 感 柑 歡 環 甘 筒 肝

540 艦 觀 貫 還 回 回 回 韓 丸 岸 眼 岩 雁 頑 顔 願 伎 喜 穿 希

560 机期機瓦祈季紀記貴鬼龜儀擬疑義誼菊吉橘詰
 580 黍容丘久休及吸宮急朽求級給旧牛去許魚京供
 600 共協境強悲恭教橋胸薈鏡驚凝業局極玉筋謹
 620 金吟銀九句区狗若具空串栗君董群軍頃刊兌啓
 640 形慶敬桂溪計詣鷄芸迎鯨欠決穴結月件健劍建
 660 兕賢軒僮元原巖源玄現言個古固己户祐湖胡虎
 680 五午後御橋語交候光厚口向好孝工幸広康恆徑
 700 校江甲紅考荒行醜降香高合克告國酷黑瀝脾胃
 720 込頃今婚恨根左砂座最妻才裁歲祭細葉際在材
 740 坂崎作昨秘鮭察猗晒三叁山散産酸殘仕伺使刺
 760 司史四士始姿子市師思指支旨枚氏私系紙紫至
 780 詞試賜雌率倚兕字寺措崎次治耳自麈式宗七失
 800 室寔芝縮含射煮社者謝車取守千朱趣酒首受壽
 820 叔周宗州秀秋終舟位充十柔汁縱重宿祝縮出叵

840 俊春旬潤純順處初所暑署書諸助女除傷高將小
 860 少抄昭晶松消燒照証賞醬鐘障上大乘城場孃常
 880 情奈狀蒸釀植色會信心振新深申真神自親身辛
 900 隄針人仁塵去尋尽酢吹水炊醉隨教杉雀世瀨制
 920 勢性成望晴正清生盛精声製西請青稅席惜者石
 940 赤跡切折設節雪絕舌蟬仙先千占專川戰扇泉淺
 960 沈滌選鮮爾然全禪膳祖素組僧喪壯掃早曹菓相
 980 草藻裝走送霜增臟藏贈造鼻束俗賊族統存尊他
 1000 多太打体对侍戴苔袋鯛代台大第題沢茸只僅谷
 1020 鱈舟卑担旦淡炭短胆团壇彈暖檀段男值知地智
 1040 池置匠竹筑茶肴中仲忠柱注虫耐猪丁帳張朝潮
 1060 町賜調長頂鳥直玲陳痛匱塚漬爪鶴亭低貞定帝
 1080 茅程泥的撤鉄典天店添甜軫点佗殿田斗社登都
 1100 度土怒冬凍刀唐鳥投東桃湯当答筒糖統藤豆頭

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

1120 勳同堂尊童菊道得德特独詭豚奈内風鍋南難二
1140 式句肉日乳入如菲任忍認寧蔥熟年念乃之納把
1160 波馬廢祥排杯配倍梅壳稔刺博白粕薄麦箱八登
1180 醜拔鳩半帆板繁般飲晚否彼悲比皮避非飛枇美
1200 必筆姪百標冰表評痼秒苗品不付夫婦富布府晉
1220 淳父腐負武舞葡莖部封風落伏復服福私仙物分
1240 粉文圓兵平柄並困米別變片編凹區便弁保步鞞
1260 暮毋簿匄報奉宝放方泥烹蜂訪亡忘房望棒鋒北
1280 墨睦没本凡盆麻妹枚每幕枕鮪又未万慢味未密
1300 密妙民務萼無娘冥名命明迷鳴免面麵毛猛網木
1320 目庆伺门夜野矢役藥藪油勇友有袖由裕甌雄夕
1340 預揭曜樣洋用羊葉耍踊陽養浴羅来賴络落卵嵐
1360 覽利梨理里陸立略流留竟龍慮兩料涼稜良力綠
1380 林琳輪令例冷礼鈴曆歷戀練蓮逕路露朗老郎六

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

1400 禄 和 話 詫 井 會 姜 宗 屏 狂 揉 撻 朧 條 椒 榨 檬 澤 炒 焙

1420 燻 炯 珈 瑰 疣 菝 茹 蝮 蛭 鬻 證 貽 踴 辣 頌 餃 餡 饅 鮑 鮫

1440 鯉 鰈 鯨 鷄 麩 飴 鱈 晦 葛 祇 薯 匪 增 挽 餅 葡 兔 師 釜 鞭

1460 / > \ ！ " # \$ % & ' () * + , - . / 0

1480 1 2 3 4 5 6 7 8 9 : ; < = > ? @ A B C D

1500 E F G H I J K L M N O P Q R S T U V W X

1520 Y Z [\] ^ _ ` a b c d e f g h i j k l

1540 m n o p q r s t u v w x y z { | } ~ … %

1560 ' , " ” _ — • — ‘ ’ ‘ ’ € £ \$ %

1580 © • ° ½ ⅓ ⅔ ¼ ¾ ⅛ ⅜ ⅝ ⅞ ⅺ ± × + − √ ∕ ∓

1600 ∘ ∙ ∘ ∙ ∘ ∙ ∘ ∙ ∘ ∙ ∘ ∙ ∘ ∙ ∘ ∙ ∘ ∙ ∘ ∙ ∘ ∙ ∘ ∙ ∘ ∙

1620 ; ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’

1640) (‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’ ‘ ’

1660 あ ぁ い い ぅ ぅ え え お お か か け け こ こ

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1680	さ	ざ	し	じ	す	ず	せ	ぜ	そ	ぞ	た	だ	ち	ぢ	っ	づ	つ	づ	て	て
1700	ど	な	に	ぬ	ね	の	は	ば	ひ	び	ひ	よ	よ	へ	べ	へ	ほ	ほ		
1720	ほ	ま	み	む	め	も	や	や	ゆ	ゆ	よ	よ	り	り	る	わ	わ	わ	わ	る
1740	を	を	ん	う	か	け	ア	ア	イ	イ	ウ	ウ	エ	エ	オ	オ	カ	ガ	キ	ギ
1760	ク	ケ	ケ	コ	ゴ	サ	ザ	シ	ジ	ス	ズ	セ	ゼ	ソ	ゾ	タ	ダ	チ	チ	
1780	ツ	ツ	ヅ	テ	テ	ト	ド	ナ	ニ	ヌ	ネ	ノ	ハ	バ	パ	ヒ	ビ	ピ	フ	ブ
1800	プ	ヘ	ベ	ペ	ホ	ボ	ポ	マ	ミ	ム	メ	モ	ヤ	ヤ	ユ	ユ	ヨ	ヨ	リ	リ
1820	ル	レ	ロ	ワ	ワ	ホ	エ	シ	ン	ウ	カ	ケ	グ	ゴ	ゴ	哀	愛	換	葵	
1840	梶	鯨	圧	扱	鮎	粟	安	暗	策	杏	似	伴	位	依	困	意	異	移	胃	衣
1860	井	育	郁	磯	一	壺	稻	芋	允	員	引	飲	蔭	院	隱	右	宇	鳥	羽	兩
1880	卯	丑	鰻	涌	瓜	宮	管	歎	榮	永	英	液	趣	榎	内	園	宴	炎	煙	遠
1900	塩	於	央	輿	横	王	翁	黄	尾	桐	牡	乙	温	穩	音	下	化	何	何	何
1920	加	可	夏	家	暇	果	歌	河	火	花	茄	荷	華	菓	蝦	過	霞	我	茅	賀
1940	介	会	解	回	塊	壤	快	恠	懷	改	海	灰	界	皆	給	芥	蟹	用	階	貝

1960 外害街垣柿谷松格設確角堂樂樹竺歛割活滑且
 1980 鯉株蒲鴨粥冠寒卷完官干感柑歡環甘筒肝艦觀
 2000 貧還回回圓圓韓丸岸眼岩雁頑顏願伎喜寄希机期
 2020 機瓦析季紀記貴鬼龜儀擬疑義誼菊古橘詰黍容
 2040 丘久休及吸宮急朽求級給旧牛去許魚京供共協
 2060 境強恐恭教橋胸薈鏡驚凝業局極玉筋謹匠金吟
 2080 銀九句区徇若具空串栗君董群軍傾刑兇啓形慶
 2100 敬桂溪計詣鷄芸仰鯨欠決穴結月件健劍建見賢
 2120 軒僅元原蔽源玄現言個古固己户枯湖胡虎五午
 2140 後御橋語交候光厚口向好孝工幸広康恒控校江
 2160 甲紅考荒行醜降香高合克告國酷墨灑脾胃凶頃
 2180 今婚恨根左砂座最妻才款歲祭細葉際在材坂崎
 2200 作昨櫻鮭察得晒三叁山散產酸殘仕伺使刺司史
 2220 口士始姿子市師思指支旨枚氏私糸紙紫至詞試

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

2240 賜雌奉倚兒字寺荷峙次治耳自鹿式宗七失室冥
2260 芝縞合射盞社者謝車取守千朱趣酒首受壽叔周
2280 宗州秀秋終舟住充十柔汁縱重宿祝縮出匪俊春
2300 旬潤純順処初所暑署書諸助女除傷高樛小少抄
2320 昭晶松浦燒照証賞嚮鐘障上大乘城場孃常情条
2340 狀蒸釀植色食信心振新深申真神召親身幸進針
2360 人仁塵去尋尽醉吹水焰醉隨教於雀世瀨制勢性
2380 成望晴正清生盛精声製西請青稅席惜者石赤跡
2400 切折設節雪絕舌蟬仙先千占專川戰扇泉淺洗滌
2420 選鮮爾然全禪膳祖素組僧喪壯掃早曹巢相草藻
2440 裝走送霜增臟歲贈造鼻束俗賊族統存尊他多太
2460 打体对侍戴苔袋鯛代台大第題沃茸只達谷鱈母
2480 單袒且淡炭短胆团壇彈暖檀段男值知地智池置
2500 匪竹筑茶着中仲忠柱注虫耐猪丁帳張朝潮町腸

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

2520 調長頂鳥直玲陳痛通塚漬爪鷓亭低負定帝第程

2540 泥的徹鉄曲天店添甜軋点佗殿田斗杜登都度土

2560 怒冬凍刀唐鳥役東桃湯当答筒糖統藤豆頭勳同

2580 堂尊童菊道悒德犄独詭豚奈内風鍋南難二式句

2600 肉日乳入如菲任忍認寗葱熟年念乃之納把波馬

2620 廢样排杯配信梅壳稔刺博白柏薄麦箱八瓮醜拔

2640 鳩半帆板繫般飯晚否彼悲比皮避非飛柶美必筆

2660 姪百標冰表評病秒苗品不付夫婦富布府晉淳父

2680 腐負武舞葡莖部封風落伏復服福私仙物分粉文

2700 圓兵平柄並困米別變片編辺返便弁保步輻幕毋

2720 簿匱報奉宝放方泥烹蜂訪亡忘房望棒鋒北墨睦

2740 没本凡盆麻妹枚每幕枕鮪又未万慢味未密蜜妙

2760 氏務葶無娘冥名命明迷鳴兎面麵毛猛網木目庆

2780 伺門夜野矢役藥藪油勇友有袖由裕遊雄夕預揚

2800 曜 样 洋 用 羊 葉 象 踊 陽 養 浴 羅 來 賴 絡 落 卵 嵐 覽 利

2820 梨 理 里 陸 立 略 沆 留 竜 龍 慮 兩 料 涼 稜 良 力 綠 林 琳

2840 輪 令 例 冷 礼 鈴 曆 歷 恋 練 蓮 連 路 露 朗 老 郎 六 祿 和

2860 話 詫 井 會 姜 宗 屏 狂 揉 榷 隴 條 椒 樟 椽 澤 炒 焙 燻 烟

2880 珈 琲 疣 苺 茹 蠣 蛭 菴 證 貽 踞 辣 頌 餃 餡 饅 鮓 鱈 鰓 鰈

2900 鯨 鶉 麩 飴 鰯 晦 葛 祇 薯 匪 噌 攪 斫 捕 兎 師 釜 鞭 丿 丿

2920 丿

2940 丿

2960 丿