Source Han Sans Version 1.002

Built on March 18, 2015 & April 7, 2015 • Released on April 20, 2015

Overview

Source Han Sans, designed by Ryoko Nishizuka (西塚涼子), is a sans serif Pan-CJK font family that is offered in seven weights—ExtraLight, Light, Normal, Regular, Medium, Bold, and Heavy—and in several OpenType/CFF-based deployment configurations to accommodate various system requirements or, in some cases, limitations. Pan-CJK fonts such as Source Han Sans are intended to render the most important characters for Simplified Chinese, Traditional Chinese, Japanese, and Korean.

The samples on this page demonstrate that the differences for each language can be subtle or striking, depending on the ideograph, yet they all clearly share the same typeface design, weight, and other characteristics that are not necessarily tied to a particular language.

The first sample below shows, from left to right, the Simplified Chinese, Traditional Chinese, and shared Japanese/Korean forms of U+9AA8 in Source Han Sans:

The second sample shows, again from left to right, the Simplified Chinese, Traditional Chinese, Japanese, and Korean forms of U+66DC:

The use of these fonts is covered under the terms of the SIL Open Font License, Version 1.1.

The pages that follow provide technical details about the font resources that are included in this open source project, and correspond to the Version 1.002 release.

Configurations

Source Han Sans is provided in four basic deployment configurations, each of which is described below, along with typical usage scenarios:

Language-specific OpenType/CFF (OTF)—36 font resources

This deployment configuration is available in four languages—Simplified Chinese, Traditional Chinese, Japanese, and Korean—and sets the language as the default (a default language is required due to the single 'cmap' table), and the 'locl' (*Localized Forms*) GSUB feature is expected to be used to access glyphs that are appropriate for the other three languages. (Note that the Regular and Bold weights include two forms of the fonts for all four language, which differ only in that the default glyphs for ASCII (U+0020 through U+007E), U+00A0 ( ), U+00A5 (¥), U+2011 (-), and U+20A9 (₩) are proportional or half-width. The fonts for the latter include the additional “HW” identifier in their names.)

These fonts represent the most compact form that supports all languages and includes the complete set of glyphs, but this comes at the expense of requiring an application to properly support the 'locl' GSUB feature in order to display in languages other than the default one. In addition to using such an application, a good example of which is Adobe InDesign, the text—at the character, paragraph, or document level—must also be properly language-tagged.

OpenType/CFF Collection (OTC)—7 font resources

This deployment configuration represents a “best of all possible worlds” in that there are separate font instances for each language, and while each font instance necessarily specifies a default language, the 'locl' GSUB feature can still be used to access the glyphs for the other languages. The Regular and Bold weights additionally include font instances for all four languages whose ASCII characters are half-width instead of the usual proportional.

These fonts offer the greater flexibility in that there is a single font resource that includes four or eight font instances each with a different one of the four languages serving as its default. Users of these fonts simply choose the appropriate font in an application’s font menu, and the glyphs that are suitable for that language are displayed. However, OpenType/CFF Collections are not yet broadly supported. For example, Windows OS does not yet support them. OS X Version 10.8, iOS7, and Adobe CS6 applications represent the first environments that support this particular font format. Note that if you install the OTCs, you cannot install any of the corresponding language-specific OTFs because they share the same PostScript and Family names.

Super OpenType/CFF Collection (Super OTC)—a single font resource

This deployment configuration packs all seven weights, all four languages, and half-width variations (ASCII-only) for two of the seven weights into a single font resource that includes a total of 36 font instances and 458,745 total glyphs. Due to OpenType table sharing, there are seven unique 'CFF ', 'GPOS', 'hmtx', and 'vmtx' tables (one per weight), four unique 'GSUB' tables (one per language), and eight unique 'cmap' tables (one per language plus proportional/half-width combination). These represent the largest OpenType tables, so greater sharing leads to a smaller overall footprint. This saves over 10MB compared to the seven separate OTCs. While each font instance specifies a default language, the 'locl' GSUB feature can still be used to access the glyphs for the other languages.

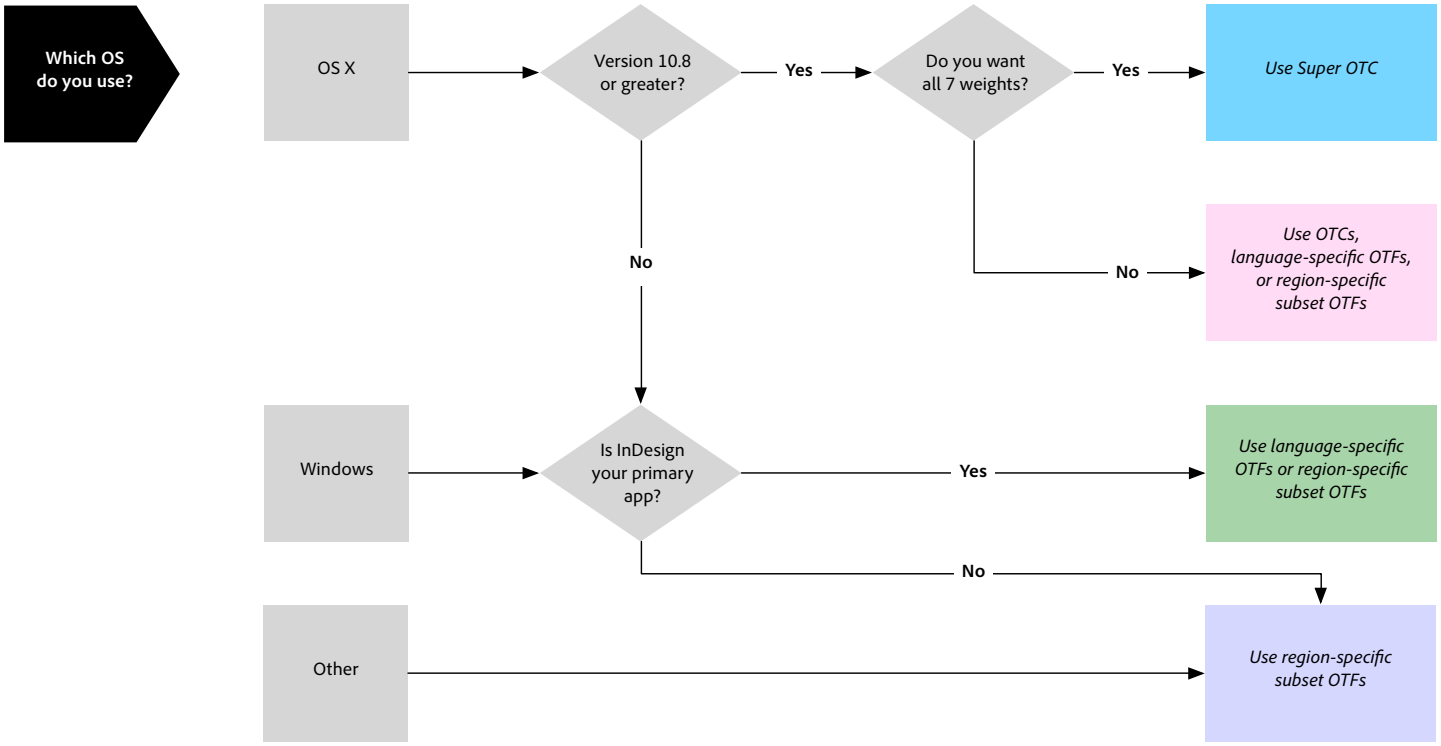
This font offers the greatest flexibility in that there is a single font resource that includes all 36 font instances, one for each of the seven weights and four languages, along with half-width ASCII variations of the Regular and Bold weights. Users of this font simply choose the appropriate font in an application’s font menu, and the glyphs that are suitable for that language are displayed. The Super OTC is subject to the same caveats and limitations as the standard OTCs.

Region-specific Subset OpenType/CFF (Subset OTF)—28 font resources

This deployment configuration includes four different subsets, and each subset includes only the glyphs that are necessary for Simplified Chinese, Traditional Chinese, Japanese, or Korean.

> *These fonts are considered the most broadly usable because the 'locl' GSUB feature is not required to access the region-specific glyphs. Instead, only the glyphs that are necessary for each region are included. This deployment configuration is recommended for users who need only the glyphs for a specific region, and also desire the smallest possible footprint. These fonts are expected to behave the same as conventional Simplified Chinese, Traditional Chinese, Japanese, or Korean fonts.*

The flowchart below may be helpful in deciding which deployment configurations are usable in your current or preferred working environment:



If the flowchart suggests that you are able to make use of more than one deployment configuration, please re-examine the descriptions above to determine which one would better satisfy your usage needs.

Our Recommendation

If the flowchart indicates that you are able to make use of more than one deployment configuration, we recommend one or more of the region-specific subset OTFs mainly due to the fact that they are usable in a broader range of environments. If you choose to download and install the language-specific OTFs, OTCs, or Super OTC, please be mindful of their requirements for accessing region-specific glyphs and various OS limitations.

Font Resources

The table below lists all 72 font resources that are included in this release, organized by format and language, and providing their file and PostScript names:

Format	Language	File Name	PostScript Name/Names
OTF	Simplified Chinese	*SourceHanSansSC-ExtraLight.otf*	SourceHanSansSC-ExtraLight
		SourceHanSansSC-Light.otf	SourceHanSansSC-Light
		SourceHanSansSC-Normal.otf	SourceHanSansSC-Normal
		SourceHanSansSC-Regular.otf	SourceHanSansSC-Regular
		SourceHanSansHWSC-Regular.otf	SourceHanSansHWSC-Regular
		SourceHanSansSC-Medium.otf	SourceHanSansSC-Medium
		SourceHanSansSC-Bold.otf	SourceHanSansSC-Bold
		SourceHanSansHWSC-Bold.otf	SourceHanSansHWSC-Bold
		SourceHanSansSC-Heavy.otf	SourceHanSansSC-Heavy
	Traditional Chinese	*SourceHanSansTC-ExtraLight.otf*	SourceHanSansTC-ExtraLight
		SourceHanSansTC-Light.otf	SourceHanSansTC-Light
		SourceHanSansTC-Normal.otf	SourceHanSansTC-Normal
		SourceHanSansTC-Regular.otf	SourceHanSansTC-Regular
		SourceHanSansHWTC-Regular.otf	SourceHanSansHWTC-Regular
		SourceHanSansTC-Medium.otf	SourceHanSansTC-Medium
		SourceHanSansTC-Bold.otf	SourceHanSansTC-Bold
		SourceHanSansHWTC-Bold.otf	SourceHanSansHWTC-Bold
		SourceHanSansTC-Heavy.otf	SourceHanSansTC-Heavy
	Japanese	*SourceHanSans-ExtraLight.otf*	SourceHanSans-ExtraLight
		SourceHanSans-Light.otf	SourceHanSans-Light
		SourceHanSans-Normal.otf	SourceHanSans-Normal
		SourceHanSans-Regular.otf	SourceHanSans-Regular
		SourceHanSansHW-Regular.otf	SourceHanSansHW-Regular
		SourceHanSans-Medium.otf	SourceHanSans-Medium
		SourceHanSans-Bold.otf	SourceHanSans-Bold
		SourceHanSansHW-Bold.otf	SourceHanSansHW-Bold
		SourceHanSans-Heavy.otf	SourceHanSans-Heavy
	Korean	*SourceHanSansK-ExtraLight.otf*	SourceHanSansK-ExtraLight
		SourceHanSansK-Light.otf	SourceHanSansK-Light
		SourceHanSansK-Normal.otf	SourceHanSansK-Normal
		SourceHanSansK-Regular.otf	SourceHanSansK-Regular
		SourceHanSansHWK-Regular.otf	SourceHanSansHWK-Regular
		SourceHanSansK-Medium.otf	SourceHanSansK-Medium
		SourceHanSansK-Bold.otf	SourceHanSansK-Bold
		SourceHanSansHWK-Bold.otf	SourceHanSansHWK-Bold
		SourceHanSansK-Heavy.otf	SourceHanSansK-Heavy
OTC	All	*SourceHanSans-ExtraLight.ttc*	SourceHanSansSC-ExtraLight, SourceHanSansTC-ExtraLight, SourceHanSans-ExtraLight, SourceHanSansK-ExtraLight
		SourceHanSans-Light.ttc	SourceHanSansSC-Light, SourceHanSansTC-Light, SourceHanSans-Light, SourceHanSansK-Light
		SourceHanSans-Normal.ttc	SourceHanSansSC-Normal, SourceHanSansTC-Normal, SourceHanSans-Normal, SourceHanSansK-Normal
		SourceHanSans-Regular.ttc	SourceHanSansSC-Regular, SourceHanSansHWSC-Regular, SourceHanSansTC-Regular, SourceHanSansHWTC-Regular, SourceHanSans-Regular, SourceHanSansHW-Regular, SourceHanSansK-Regular, SourceHanSansHWK-Regular
		SourceHanSans-Medium.ttc	SourceHanSansSC-Medium, SourceHanSansTC-Medium, SourceHanSans-Medium, SourceHanSansK-Medium
		SourceHanSans-Bold.ttc	SourceHanSansSC-Bold, SourceHanSansHWSC-Bold, SourceHanSansTC-Bold, SourceHanSansHWTC-Bold, SourceHanSans-Bold, SourceHanSansHW-Bold, SourceHanSansK-Bold, SourceHanSansHWK-Bold
		SourceHanSans-Heavy.ttc	SourceHanSansSC-Heavy, SourceHanSansTC-Heavy, SourceHanSans-Heavy, SourceHanSansK-Heavy

Format	Language	File Name	PostScript Name/Names
Super OTC	All	*SourceHanSans.ttc*	SourceHanSansSC-ExtraLight, SourceHanSansSC-Light, SourceHanSansSC-Normal, SourceHanSansSC-Regular, SourceHanSansHWSC-Regular, SourceHanSansSC-Medium, SourceHanSansSC-Bold, SourceHanSansHWSC-Bold, SourceHanSansSC-Heavy, SourceHanSansTC-ExtraLight, SourceHanSansTC-Light, SourceHanSansTC-Normal, SourceHanSansTC-Regular, SourceHanSansHWTC-Regular, SourceHanSansTC-Medium, SourceHanSansTC-Bold, SourceHanSansHWTC-Bold, SourceHanSansTC-Heavy, SourceHanSans-ExtraLight, SourceHanSans-Light, SourceHanSans-Normal, SourceHanSans-Regular, SourceHanSansHW-Regular, SourceHanSans-Medium, SourceHanSans-Bold, SourceHanSansHW-Bold, SourceHanSans-Heavy, SourceHanSansK-ExtraLight, SourceHanSansK-Light, SourceHanSansK-Normal, SourceHanSansK-Regular, SourceHanSansHWK-Regular, SourceHanSansK-Medium, SourceHanSansK-Bold, SourceHanSansHWK-Bold, SourceHanSansK-Heavy
Subset OTF	CN	*SourceHanSansCN-ExtraLight.otf*	SourceHanSansCN-ExtraLight
		SourceHanSansCN-Light.otf	SourceHanSansCN-Light
		SourceHanSansCN-Normal.otf	SourceHanSansCN-Normal
		SourceHanSansCN-Regular.otf	SourceHanSansCN-Regular
		SourceHanSansCN-Medium.otf	SourceHanSansCN-Medium
		SourceHanSansCN-Bold.otf	SourceHanSansCN-Bold
		SourceHanSansCN-Heavy.otf	SourceHanSansCN-Heavy
	TW	*SourceHanSansTW-ExtraLight.otf*	SourceHanSansTW-ExtraLight
		SourceHanSansTW-Light.otf	SourceHanSansTW-Light
		SourceHanSansTW-Normal.otf	SourceHanSansTW-Normal
		SourceHanSansTW-Regular.otf	SourceHanSansTW-Regular
		SourceHanSansTW-Medium.otf	SourceHanSansTW-Medium
		SourceHanSansTW-Bold.otf	SourceHanSansTW-Bold
		SourceHanSansTW-Heavy.otf	SourceHanSansTW-Heavy
	JP	*SourceHanSansJP-ExtraLight.otf*	SourceHanSansJP-ExtraLight
		SourceHanSansJP-Light.otf	SourceHanSansJP-Light
		SourceHanSansJP-Normal.otf	SourceHanSansJP-Normal
		SourceHanSansJP-Regular.otf	SourceHanSansJP-Regular
		SourceHanSansJP-Medium.otf	SourceHanSansJP-Medium
		SourceHanSansJP-Bold.otf	SourceHanSansJP-Bold
		SourceHanSansJP-Heavy.otf	SourceHanSansJP-Heavy
	KR	*SourceHanSansKR-ExtraLight.otf*	SourceHanSansKR-ExtraLight
		SourceHanSansKR-Light.otf	SourceHanSansKR-Light
		SourceHanSansKR-Normal.otf	SourceHanSansKR-Normal
		SourceHanSansKR-Regular.otf	SourceHanSansKR-Regular
		SourceHanSansKR-Medium.otf	SourceHanSansKR-Medium
		SourceHanSansKR-Bold.otf	SourceHanSansKR-Bold
		SourceHanSansKR-Heavy.otf	SourceHanSansKR-Heavy

Glyph Set Particulars

Glyph Set & Region-specific Subsets

The number of glyphs in each font resource—except for the region-specific subset OTFs—is 65,535 (CIDs 0 through 65534), which is at the architectural limit for CID-keyed fonts (65,535 glyphs).

The table below indicates the number of glyphs that are included in the region-specific subset OTFs, whose figures include a common set of 2,563 glyphs that represent various characters, symbols, and punctuation.

Also provided are the names of the subset definition files that are included in this open source project.

Language	Glyphs	Subset Definition File	Supported Standards
Simplified Chinese	30,907	*AI0-SourceHanSans.CN*	All GB 18030 hanzi, all 8,105 hanzi of *Tōngyòng Guīfàn Hànzìbiǎo* (通用规范汉字表) 199 of which are outside of GB 18030
Traditional Chinese	20,826	*AI0-SourceHanSans.TW*	All Big Five hanzi, all Hong Kong SCS-2008 hanzi (adhering to the Taiwan MOE glyph standard), the seven ETen hanzi
Japanese	17,850	*AI0-SourceHanSans.JP*	All Adobe-Japan1-6 kanji (a superset of those in JIS X 0208, JIS X 0213 & JIS X 0212)
Korean	24,737	*AI0-SourceHanSans.KR*	All contemporary (11,172) and 500 high-frequency archaic hangul syllables, conjoining hangul jamo (with full archaic hangul support), all KS X 1001 and KS X 1002 hanja (7,476), 466 additional hanja

Of course, the font resources that include the full set of 65,535 glyphs support all of the standards that are listed in the above table, and employ some method of accessing the glyphs for different languages when they occupy the same Unicode code point and require a different shape.

The ordering file, *AI0-SourceHanSans*, lists all 65,535 CIDs in the first column, and shows the FDArray and row font structure in the second and third columns, respectively, along with the Unicode-based working glyph names in the fourth column. All 65,535 working glyph names are unique, and all—with the exception of the ones for CID+0 (the *.notdef* glyph) and CIDs 65485 through 65534—use a “uni” (BMP) or “u” (outside BMP) prefix followed by uppercase hexadecimal digits. Glyphs that are represented by (or can be considered) sequences are made up of concatenations of the appropriate Unicode-based glyph names. Identifiers for regions and other purposes are also used.

Weights

The table below shows sample glyphs in each of the seven weights, ranging from ExtraLight to Heavy. The ExtraLight and Heavy weights represent the master designs, and the five intermediate weights are the result of multiple master interpolation (the interpolation ratios are provided):

ExtraLight—0	Light—160	Normal—320	Regular—420	Medium—560	Bold—780	Heavy—1000
汉漢 漢한	汉漢 漢한	汉漢 漢한	汉漢 漢한	汉漢 漢한	汉漢 漢한	汉漢 漢한

Glyph Complement PDFs

Included in this open source project are seven Unicode-based glyph complement PDFs, one for each weight, that provide a visual synopsis of the UTF-32 'cmap' tables for each of the four languages: Simplified Chinese, Traditional Chinese, Japanese, and Korean. For each code point that maps to a glyph, there are three types of annotations, described as follows according to their position relative to the code-point box:

Upper-Left—Glyph width: **F** = Full-width, **H** = Half-width, **M** = Monospaced (hangul letters and syllables), **P** = Proportional, **Q** = Quarter-width, **T** = Tall (U+3031, U+3032, and the vertical forms of U+2E3A and U+2E3B), **W** = Wide (U+2E3A and U+2E3B), **Z** = Zero (non-spacing)

Upper-Right—Language/Region: **C** = Simplified Chinese (China), **H** = Traditional Chinese (Hong Kong SAR), **J** = Japanese, **K** = Korean, **T** = Traditional Chinese (Taiwan)

Bottom—The CID of the glyph

Each glyph complement PDF contains four bookmarked 384-page sections, one for each language, meaning 1,536 pages in total. Glyphs that are tall (T), wide (W), or non-spacing (Z) may exceed or appear outside the code-point box, which includes those for U+2E3A, U+2E3B, U+302A through U+302D, U+3031, and U+3032.

Also included in this open source project are seven glyph complement PDFs, one for each weight, that show the 500 pre-composed high-frequency archaic hangul syllables, ordered by their two- or three-character combining sequences.

Unencoded Glyphs

Not shown in the Unicode-based glyph complement PDFs are glyphs that are unencoded.

Ignoring code points that share different Simplified Chinese, Traditional Chinese, Japanese, Korean, and proportional/half-width glyphs, there are 3,326 unencoded glyphs in each 65,535-glyph font resource. The region-specific subset OTFs include considerably fewer unencoded glyphs.

Approximately one-third of the unencoded glyphs are Japanese ideographs (kanji), all of which represent kanji included in Adobe-Japan1-6. Some of these have been explicitly identified as JIS90 (JIS X 0208-1990) glyphs according to their source glyph names (CIDs 61783 through 61951; 169 glyphs) and are reflected in the 'jp90' GSUB feature that is specific to Japanese fonts and font instances, and the remainder have been identified according to their registered IVSes (CIDs 61952 through 62994; 1,043 glyphs) in the *Adobe-Japan1* IVD (Ideographic Variation Database) Collection, and are reflected in the Format 14 'cmap' subtable in the same fonts and font instances.

The bulk of the remaining unencoded glyphs are the 500 high-frequency archaic hangul syllables, the glyphs for combing jamo, vertical forms, and a small number of other variants.

Latin, Greek & Cyrillic Glyphs

Included in all font resources is a rich set of Latin glyphs that support not only ASCII and ISO/IEC 8859-1 (aka ISO Latin 1), but also the characters that are necessary for broadly-used CJK transliteration and transcription systems, along with those that are necessary for Latin-based Vietnamese. A basic set of glyphs for Greek and Cyrillic, with proportional metrics, is also included.

Source Han Sans Versus Source Sans Pro & Source Code Pro

The Latin, Latin-like, Greek, and Cyrillic glyphs in Source Han Sans are derived from—but not identical to—Source Sans Pro. The same is true for the half-width glyphs in terms of their relationship with Source Code Pro. The Latin and Latin-like glyphs in a typical CJK font represent a minority, and when it comes to harmonizing glyphs of different scripts, it is better to modify the minority to harmonize with the majority, and not vice versa. In addition, half-width glyphs in typical CJK fonts are also expected to be precisely half-width.

There are two primary differences between the glyphs that are common in Source Han Sans and Source Sans Pro:

- The interpolation ratios for the weights are different. Source Han Sans is available in seven weights: ExtraLight, Light, Normal, Regular, Medium, Bold, and Heavy. Source Sans Pro is available in six: ExtraLight, Light, Regular, Semibold, Bold, and Black. While some of the weight names are the same, one should not expect that the interpolation ratios are the same. They will be relatively close, but not precisely the same.
- The glyphs in Source Han Sans that are derived from Source Sans Pro have been adapted for use in Source Han Sans, which mainly involves scaling. In the case of the ExtraLight, Regular, and Heavy/Black weights, the Source Sans Pro glyphs were scaled to 110%, 113%, and 115%, respectively. Thus, the Source Han Sans glyphs appear to be slightly larger than those in Source Sans Pro, particularly in the heavier weights.

The half-width Latin glyphs in Source Han Sans, which are the default for the half-width (“HW”) OTFs and OTC font instances in only the Regular and Bold weights, and which are also exposed via the 'hwid' GSUB feature in the other OTFs and OTC font instances, are different from the glyphs in Source Code Pro in the following ways:

- Like Source Sans Pro, the interpolation ratios are different for all weights.
- The half-width Latin glyphs in Source Han Sans are precisely half-width, meaning half an em or 500-unit horizontal advances. The glyphs in Source Code Pro are monospaced, using 600-unit horizontal advances, meaning that they are not precisely half-width.
- The glyphs themselves are also different, particularly the one for zero (0) whose glyph in Source Han Sans lacks a center dot to more easily distinguish it from uppercase O, which is important when using a font to edit or display source code.

The table below compares Source Han Sans with Source Sans Pro and Source Code Pro for three weights, ExtraLight, Regular, and Heavy/Black:

Weight	Source Han Sans & Source Han Sans HW / 'hwid'	Source Sans Pro & Source Code Pro
ExtraLight	Unicode Version 8.0	↔ Unicode Version 8.0
	Unicode Version 8.0	↔ Unicode Version 8.0
Regular	Unicode Version 8.0	↔ Unicode Version 8.0
	Unicode Version 8.0	↔ Unicode Version 8.0
Heavy/ Black	**Unicode Version 8.0**	↔ **Unicode Version 8.0**
	Unicode Version 8.0	↔ **Unicode Version 8.0**

Vertical Glyphs

The usual and expected set of vertical glyphs is included, some of which are region- or language-specific. In addition, all glyphs for kana, meaning not only those for small kana, include a vertical glyph variant.

The pre-rotated non–full-width glyphs that are typically accessible via the effectively-deprecated 'vrt2' GSUB feature have been intentionally excluded from the glyph set.

CIDFont Resource & CFF Particulars

CIDFont Resource Structure

The font resources that include 65,535 glyphs began their life as an *Adobe-Identity-0* ROS CIDFont resource that includes 19 FDArray elements, each of which specifies its own hinting parameters. The table below shows the names of each of the 19 FDArray elements, its index, the CIDs and CID ranges that are included, and the total number of glyphs:

FDArray Name	Index	CIDs & CID Ranges	Glyphs
Alphabetic	0	59024–59049, 59056–59081	52
AlphabeticDigits	1	949–968, 59007–59016, 59214–59224	41
Bopomofo	2	1642–1682, 1792–1818, 65353	69

FDArray Name	Index	CIDs & CID Ranges	Glyphs
Dingbats	3	102, 112, 117, 150, 182, 246–250, 713, 716–717, 724–725, 727–730, 735, 737, 741, 744–751, 754, 756–758, 760–761, 786–948, 969–1068, 1229–1276, 1278–1283, 1285–1319, 1321–1345, 1377–1452, 1539–1542, 1546, 1637, 1776–1791, 1871–2429, 58924–59006, 59017–59023, 59050–59055, 59082–59086, 59201–59207, 59225–59313, 59316–59451, 63024–63037, 63153–63156, 65145–65165, 65252–65255, 65259, 65370–65470	1,548
DingbatsDigits	4	762–785	24
Generic	5	0, 1069–1228, 1277, 1284, 65506–65534	192
HDingbats	6	59208–59213	6
HHangul	7	59150–59200	51
HKana	8	59087–59149	63
HWidth	9	63039–63054, 63065–63135	87
HWidthCJK	10	63136–63152	17
HWidthDigits	11	63055–63064	10
Hangul	12	365–620, 1683–1775, 47537–58809, 63157–65144	13,610
Ideographs	13	1348–1376, 1819–1854, 2430–47536, 58810–58918, 59452–61768, 61783–62994	48,810
Kana	14	1453–1538, 1543–1545, 1547–1636, 1638–1641, 1855–1870, 61769–61782, 65486–65495	223
Proportional	15	1–16, 27–101, 103–111, 113–116, 118–149, 151–181, 183–245, 251–364, 621–712, 714–715, 718–723, 726, 731–734, 736, 738–740, 742–743, 752–753, 755, 759, 1320, 1346–1347, 58919–58923, 59314–59315, 63038, 65485	471
ProportionalCJK	16	62995–63023	29
ProportionalDigits	17	17–26	10
VKana	18	65166–65251, 65256–65258, 65260–65352, 65354–65369, 65471–65484, 65496–65505	222

CFF Subroutinization

All 'CFF ' tables have been subroutinized. The size savings range anywhere from 1.5MB to 3MB for the 65,535-glyph OTFs and OTCs. The ExtraLight weight exhibits the greatest size savings. Depending on the machine that is used, in terms of processor and available memory, it can take several hours to subroutinize a single 65,535-glyph 'CFF ' table. The 'CFF ' tables of the region-specific subset OTFs take considerably less time to subroutinize.

A special AFDKO *makeotf* command-line option and argument, *-maxs 30000*, was specified to keep the number of subroutines under the “32K − 3” (32,765) threshold (the implementation limit is “64K − 3” subroutines, though some environments may treat “32K − 3” as the actual limit).

It is important to understand that limiting the number of subroutines did not, in any practical or meaningful sense, cripple the 'CFF ' tables in terms of their potential to become smaller. This is because the subroutinizer identifies—early on during its processing—the most size-effective subroutines, and at some point during the processing the size of the subroutine data may cancel out—or nearly cancel out—the size-savings that is achieved through subroutinization. Exhaustive testing revealed that the 30,000-subroutine limit that has been imposed is already at or near the point of diminishing returns.

Unicode Particulars

Unicode Mappings

The Format 12 (UTF-32) 'cmap' subtable of each language-specific OTF and OTC specifies 44,651 meaningful mappings, and the region-specific subset OTFs obviously include less. Note that some glyphs map from multiple code points, such as the entire U+2F*xx* range, along with a large chunk of the CJK Compatibility Ideographs. When the eight UTF-32 CMap resources are combined, a total of 62,209 glyphs are covered, which leaves 3,326 glyphs as being not directly unencoded. Four of the mappings in the URO (*Unified Repertoire & Ordering*), U+9FCD through U+9FD0, are expected to be included in Unicode Version 8.0 (mid-2015) but have been deemed stable enough to use.

In addition to the ideographs for which there are obviously a large number of language-specific glyphs, the following code points also exhibit language-specific variation:

Unicode	Simplified Chinese	Traditional Chinese	Japanese	Korean
U+2018	‘	‘	‘	‘
U+2019	’	’	’	’
U+201C	“	“	“	“
U+201D	”	”	”	”
U+2264	≤	≤	≤	≤
U+2265	≥	≥	≥	≥
U+226E	≮	≮	≮	≮
U+226F	≯	≯	≯	≯
U+3001	、	、	、	、
U+3002	。	。	。	。
U+FF01	！	！	！	！
U+FF0C	，	，	，	，
U+FF0E	．	．	．	．
U+FF1A	：	：	：	：
U+FF1B	；	；	；	；
U+FF1F	？	？	？	？

Included as part of this open source project are the raw (and human-readable) UTF-32 mapping files—named *utf32-cn.map*, *utf32-tw.map*, *utf32-jp.map*, and *utf32-kr.map*—that are used to compile the UTF-32 CMap resources—named *UniSourceHanSansCN-UTF32-H*, *UniSourceHanSansTW-UTF32-H*, *UniSourceHanSansJP-UTF32-H*, and *UniSourceHanSansKR-UTF32-H*—that the AFDKO *makeotf* tool uses to

generate the Format 12 (UTF-32) 'cmap' subtables. Also included in this project are the raw UTF-32 mapping files—named *utf32hw-cn.map*, *utf32hw-tw.map*, *utf32hw-jp.map*, and *utf32hw-kr.map*—that are used to compile the UTF-32 CMap resources that map ASCII (U+0020 through U+007E), U+00A0 ( ), U+00A5 (¥), U+2011 (-), and U+20A9 (₩) to half-width forms—named *UniSourceHanSansHWCN-UTF32-H*, *UniSourceHanSansHWTW-UTF32-H*, *UniSourceHanSansHWJP-UTF32-H*, and *UniSourceHanSansHWKR-UTF32-H*.

Matching UTF-16 CMap resources, which should not be used to build the OpenType fonts, are included for good measure.

Unicode Coverage

In addition to complete URO, Extension A, and contemporary hangul syllable coverage, the 65,535-glyph font resources completely cover the following 256-character Unicode blocks: U+00*xx*, U+11*xx*, U+2F*xx* through U+33*xx* (except for U+332C), U+D7*xx*, U+FF*xx*, and U+1F2*xx*.

Unicode Variation Sequences

All registered *Adobe-Japan1* IVSes—except for <6CE8 E0102> (Adobe-Japan1-6 CID+12869), which is excluded because it is outside the scope of the Source Han Sans glyph set—are specified in the Format 14 'cmap' subtable of each Japanese font and font instance, along with 89 of the 1,002 Standardized Variants that were introduced in Unicode Version 6.3. This means that 14,678 *Adobe-Japan1* IVSes and 89 Standardized Variants are included. 13,307 of these UVSes are default, meaning that the glyph is directly encoded, and the remaining 1,460 are non-default (unencoded or encoded in a CJK Compatibility Ideograph block, at least for Japanese fonts and font instances). The provided *SourceHanSans_JP_sequences.txt* file specifies the UVSes.

Each Korean font and font instance includes 270 UVSes that correspond to 270 of the 1,002 Standardized Variants that were introduced in Unicode Version 6.3. All of these UVSes are default. The provided *SourceHanSans_KR_sequences.txt* file specifies the UVSes.

Each Traditional Chinese font and font instance includes 14 UVSes that correspond to 14 of the 1,002 Standardized Variants that were introduced in Unicode Version 6.3. Two of these UVSes are default (directly encoded), and the remaining 12 are non-default (encoded in a CJK Compatibility Ideograph block). The provided *SourceHanSans_TWHK_sequences.txt* file specifies the UVSes.

The Simplified Chinese fonts and font instances do not need to include a Format 14 'cmap' subtable.

UTR #50 Compliance

Source Han Sans is one of the first font implementations that is compliant with UTR #50 (*Unicode Vertical Text Layout*). Only the substitutions in the 'vert' GSUB feature are expected to be used, and the 'vrt2' GSUB feature, which is a subset of the 'vert' GSUB feature, is included only because some environments, such as Windows and some Microsoft applications, require it to be present. In particular, pre-rotated non–full-width glyphs have been excluded from the 'vrt2' GSUB feature, and substitutions for arrows and arrow-like characters have also been excluded from both GSUB features.

Language Particulars

Simplified Chinese: GB 18030 & China's Tōngyòng Guīfàn Hànzìbiǎo

In addition to supporting GB 18030, which primarily amounts to Simplified Chinese glyphs for all URO and Extension A code points plus six Extension B code points, China's latest list of 8,105 hanzi (通用规范汉字表 *Tōngyòng Guīfàn Hànzìbiǎo*), which includes 196 additional Extension B through E code points, along with

three to be appended to the URO for 199 in total, is also supported. Among these 199 hanzi, 36 map to Extension B, 44 map to Extension C, eight map to Extension D, 108 map to Extension E, and three have been appended to the URO (U+9FCD through U+9FCF).

Traditional Chinese: TW Versus HK Coverage

Beginning with Version 1.001, there will be separate TW (Taiwan) and HK (Hong Kong SAR) fonts and font instances for Traditional Chinese, with the combined TW and HK fonts and font instances from Version 1.000 serving as the former, because the glyphs continue to adhere to the Taiwan MOE (Ministry of Education) glyph standard. The latter, meaning the HK fonts, will be released at a later date, in an experimental form that will repurpose existing glyphs to the extent that is possible, at least initially.

Due to the scope of Traditional Chinese coverage, which is limited to Big Five (equivalent to CNS 11643 Planes 1 and 2) and Hong Kong SCS, any CJK Unified Ideograph code point that is outside that scope is not likely to display appropriately for Traditional Chinese use.

Japanese: Adobe-Japan1-6 Correspondence Table & JIS Coverage

The provided *aj16-kanji.txt* mapping file shows how all Adobe-Japan1-6 kanji map to working glyph names as specified in the fourth field of the included *AI0-SourceHanSans* ordering file. In order to support the *Adobe-Japan1* IVD Collection, glyphs for all Adobe-Japan1-6 kanji—except for <6CE8 E0102> (Adobe-Japan1-6 CID+12869), which is excluded because it is outside the scope of the Source Han Sans glyph set—are included.

Due to the JIS standard coverage of Adobe-Japan1-6 that is inherited by Source Han Sans, all JIS X 0208, JIS X 0213, and JIS X 0212 kanji are therefore supported. JIS2004 (aka JIS X 0213:2004) glyphs are the default for the relevant code points. A small number of characters in the JIS standards, such as those for IPA, along with additional Latin, Greek, and Cyrillic that were not deemed necessary, have been intentionally excluded.

Although Source Han Sans includes the same kanji as Adobe-Japan1-6, including a large number of kanji variants, compatibility shouldn't be expected for documents that were authored using applications that specify glyphs by CID. The only Adobe-Japan1-6 compatibility that should be expected is at the Unicode level, which includes the *Adobe-Japan1* IVSes that are specified in the Format 14 'cmap' subtable.

Korean: Hangul Glyphs & Hanja Coverage

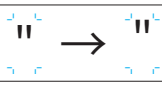
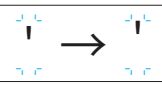
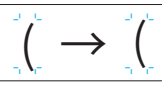
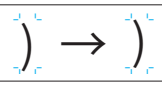
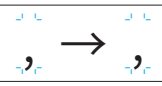
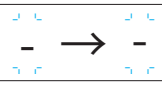
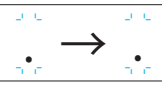
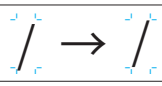
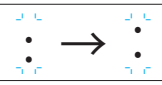
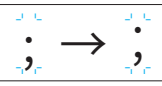
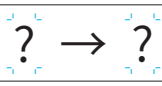
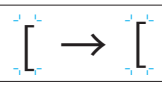
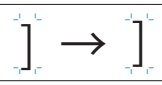
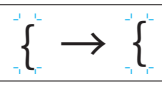
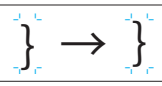
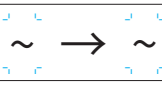
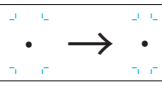
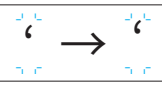
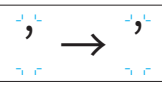
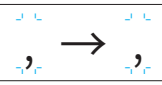
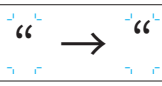
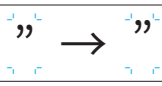
Glyphs for all contemporary Korean hangul symbols, letters (including compatibility and half-width versions), and syllables are included, along with the additional glyphs necessary to compose archaic hangul via the 'ljmo', 'vjmo', and 'tjmo' GSUB features. Also included are glyphs for 500 high-frequency archaic hangul syllables in pre-composed form, which are made accessible via the 'ccmp' GSUB feature.

While the horizontal advances of the glyphs for Korean hangul symbols (in the U+32*xx* block) are full-width (1000 units), those for Korean hangul letters and syllables are monospaced at 920 units.

The provided *ks-hanja.txt* mapping file shows how the hanja in the KS X 1001 (4,620) and KS X 1002 (2,856) standards map to working glyph names as specified in the fourth field of the included *AI0-SourceHanSans* ordering file.

Proportional & Half-Width CJK Punctuation

Included in these fonts are special forms of the proportional and half-width punctuation shown in the table below, which have been tailored for CJK use in that they are aligned to the em-box, not to Latin features, and which are accessible via the 'locl' GSUB feature:

Unicode	Proportional	Half-Width[1]	Chinese	Japanese	Korean
U+0021					Yes
U+0022			Yes	Yes	Yes
U+0027			Yes	Yes	Yes
U+0028					Yes
U+0029					Yes
U+002C					Yes
U+002D					Yes
U+002E					Yes
U+002F					Yes
U+003A					Yes
U+003B					Yes
U+003F					Yes
U+005B					Yes
U+005D					Yes
U+007B					Yes
U+007D					Yes
U+007E					Yes
U+00B7					Yes
U+2018			Yes[2]	Yes	Yes
U+2019			Yes[2]	Yes	Yes
U+201A			Yes	Yes	Yes
U+201C			Yes[2]	Yes	Yes
U+201D			Yes[2]	Yes	Yes
U+201E			Yes	Yes	Yes
U+203C					Yes
U+2047					Yes

Unicode	Proportional	Half-Width[1]	Chinese	Japanese	Korean
U+2048	?! → ?!				Yes
U+2049	!? → !?				Yes
U+2E3A	—— → ——		Yes	Yes	Yes
U+2E3B	——— → ———		Yes	Yes	Yes

1 The half-width glyphs are not encoded by default, except for the Regular and Bold OTFs and OTC font instances that include the “HW” identifier in their names, and are accessible via the 'hwid' GSUB feature in all other OTFs and OTC font instances .
2 The default glyph for this code point is full-width, not proportional, so the 'pwid' GSUB feature must first be invoked to access the proportional glyph that is tailored for CJK use.

OpenType Particulars

Menu Names

The table below shows the English and localized Family names for each font and font instance:

Configuration	Family Name—English	Family Name—Localized
Simplified Chinese OTF & OTC	*Source Han Sans SC*	思源黑体
	Source Han Sans HW SC	思源黑体 HW
Traditional Chinese OTF & OTC	*Source Han Sans TC*	思源黑體
	Source Han Sans HW TC	思源黑體 HW
Japanese OTF & OTC	*Source Han Sans*	源ノ角ゴシック
	Source Han Sans HW	源ノ角ゴシック HW
Korean OTF & OTC	*Source Han Sans K*	본고딕
	Source Han Sans HW K	본고딕 HW
Simplified Chinese OTF (subset)	*Source Han Sans CN*	思源黑体 CN
Traditional Chinese OTF (subset)	*Source Han Sans TW*	思源黑體 TW
Japanese OTF (subset)	*Source Han Sans JP*	源ノ角ゴシック JP
Korean OTF (subset)	*Source Han Sans KR*	본고딕 KR

For the region-specific subset OTFs, the English and localized menu names also include region identifiers. Their PostScript names, as shown in the table in the section entitled “Font Resources” section on page 3, use the same region identifiers. With the exception of Japanese, the language-specific OTFs and OTCs include a one- or two-letter language identifier only for their English menu names.

Because the OTCs and the corresponding language-specific OTFs specify identical PostScript and Family names, they cannot be installed in the same environment.

OpenType Tables

All font resources, with the exception of the OTCs, include the following 16 OpenType tables: 'BASE', 'CFF ', 'DSIG', 'GPOS', 'GSUB', 'OS/2', 'VORG', 'cmap', 'head', 'hhea', 'hmtx', 'maxp', 'name', 'post', 'vhea', and 'vmtx'. The OTCs do not include a 'DSIG' table.

The four or eight font instances in each of the seven OTCs share the following ten OpenType tables: 'BASE', 'CFF ', 'GPOS', 'VORG', 'hhea', 'hmtx', 'maxp', 'post', 'vhea', and 'vmtx'. The following five OpenType tables are not completely shared by the four or eight font instances in each OTC: 'GSUB', 'OS/2', 'cmap', 'head', and 'name'. The Super OTC shares OpenType tables more efficiently.

OpenType Table Overrides

Several values in particular OpenType tables have been overridden from their otherwise default values. The subsections below detail some of the more important table-specific overrides that have been applied.

'OS/2' Table Overrides

The *OS/2.sTypoLineGap* value has been set to 0 (zero) units, and is also reflected in the *hhea.LineGap* and *vhea.lineGap* values. The *OS/2.windowsAscent* and *OS/2.windowsDescent* values have been calculated by removing the excessively tall vertical-only glyphs—for U+2E3A, U+2E3B, U+3031, and U+3032—from the equation. This is for the benefit of applications that use these values for determining default leading. These and other 'OS/2' table settings are intended to provide consistent cross-platform line spacing (aka vertical metrics).

'name' Table Overrides

Unlike mainstream OpenType/CFF CJK fonts, a *name.ID=20* string is not specified because there are no legacy (non-Unicode) encodings that meaningfully correspond to these fonts.

'vmtx' Table Overrides

In addition to specifying alternate vertical origins for full-width Latin and Latin-like glyphs that rest on the Latin baseline, proper vertical origins and vertical advances are also specified for the glyphs that correspond to U+3031 and U+3032, and to the vertical forms of U+2E3A, U+2E3B, and U+302A through U+302F.

OpenType GSUB Features

All fonts and font instances include the OpenType GSUB features (see the OpenType Feature Registry for additional information) as detailed in the table below:

	OTF & OTC				Subset OTF			
GSUB Feature	SC	TC	J	K	CN	TW	JP	KR
aalt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
calt	Yes	Yes	Yes	Yes				Yes
ccmp	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
dlig	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
fwid[1]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
hist	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
hngl[1 2]				Yes				Yes
hwid[1]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
jp78[1]			Yes				Yes	
jp83[1]			Yes				Yes	
jp90[1]			Yes				Yes	
liga	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
ljmo	Yes	Yes	Yes	Yes				Yes
locl	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
nlck[1]			Yes				Yes	

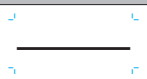
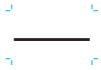
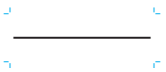
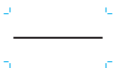
GSUB Feature	OTF & OTC				Subset OTF			
	SC	TC	J	K	CN	TW	JP	KR
pwid[1]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
tjmo	Yes	Yes	Yes	Yes				Yes
vert[1]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
vjmo	Yes	Yes	Yes	Yes				Yes
vrt2[3]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

1 This GSUB feature is inherited by the 'aalt' GSUB feature.
2 The scope of this GSUB feature is limited to the Korean versions of glyphs that correspond to the ideographs in KS X 1001 (4,620) and KS X 1002 (2,856), meaning the default forms in the Korean fonts and font instances.
3 This GSUB feature is a subset—not superset, as is usually the case—of the 'vert' GSUB feature.

All fonts and font instances that include hangul letters or syllables include a Korean-specific glyph for the *space* character (U+0020), whose width is set to 280 units for all weights. The width of the proportional glyph for the *space* character ranges from 220 units (in ExtraLight) to 229 units (in Heavy) in all fonts and font instances, except for the half-width fonts and font instances that include the “HW” identifier in their names and whose *space* glyph is 500 units. A contextual substitution in the 'calt' GSUB feature substitutes the proportional *space* glyph with the Korean-specific version only when it is surrounded by a hangul syllable (contemporary or archaic), regardless of whether it is pre-composed or combining.

The 'ccmp' GSUB Feature

The 'ccmp' GSUB feature is used to form the sequences needed to support a small number of kana, many of which are included in JIS X 0213, along with the 500 high-frequency pre-composed archaic hangul syllables, but also to handle the following additional sequences:

Unicode Sequence	Unicode	Western Glyph	CJK Glyph
<2014 2014 2014>	U+2E3B		
<2014 2014>	U+2E3A		
<2015 2015 2015>	U+2E3B		
<2015 2015>	U+2E3A		
<3033 3035>	U+3031		
<3034 3035>	U+3032		

The glyphs that result from the first four sequences can be overridden, in terms of Western versus CJK glyph style, by applying the 'locl' GSUB feature, which entails using an application that supports this OpenType feature and properly language-tagging the text. These sequences merely serve as a convenience mechanism for environments that do not support language-tagging.

The 'locl' GSUB Feature

The 'locl' GSUB feature plays a critical role in the language-specific OTFs in that it represents the *only* mechanism within the font resource for accessing the glyphs for the non-default languages. If the 'locl' GSUB feature is not supported or not properly used, the default language is used. Each non-default language is handled via

a separate lookup that is associated with the appropriate language and script, and its purpose is to mimic the 'cmap' table of the target language.

The font instances of the OTCs also include the 'locl' GSUB feature, but its presence represents an alternate method for accessing the glyphs for the non-default languages that does not involve selecting a different font instance of the OTC.

Note that in addition to using an application that supports the 'locl' GSUB feature, such as Adobe InDesign, the text must also be properly language-tagged at the character, paragraph, or document level.

Also note that all font resources, including the region-specific subset OTFs, include the 'locl' GSUB feature. For the region-specific subset OTFs that obviously do not include glyphs for the ideographs of the non-supported regions, the 'locl' GSUB feature instead operates only on a small number of glyphs for punctuation by tailoring them for CJK use. See the table in the “Proportional & Half-Width CJK Punctuation” section on page 12 for a showing of these special glyphs.

The 'vert' GSUB Feature

The 'vert' GSUB feature includes substitutions that may be different for each region or language, which apply to the following code points:

Unicode	Simplified Chinese	Traditional Chinese	Japanese	Korean
U+2018[1]	‘ → ﹁	‘ → ′	‘	‘
U+2019[1]	’ → ﹂	’ → ′	’	’
U+201C[1]	“ → ﹃	“ → ″	“	“
U+201D[1]	” → ﹄	” → ″	”	”
U+3001	、 → ︑	、	、 → ˙	、 → ˙
U+3002	。 → ︒	。	。 → °	。 → °
U+FF01	！ → ！	！	！	！ → ！
U+FF0C	， → ，	，	， → ，	， → ，
U+FF0E	． → ．	．	． → ．	． → ．
U+FF1A	： → ：	：	： → ‥	： → ：
U+FF1B	； → ；	；	；	； → ；
U+FF1F	？ → ？	？	？	？ → ？

1 To achieve the same vertical substitution effect as Traditional Chinese for Japanese or Korean, the 'fwid' GSUB feature must first be applied to change the default proportional glyphs to their full-width forms.

OpenType GPOS Features

Six GPOS features, 'halt', 'kern', 'palt', 'vhal', 'vkrn', and 'vpal', are included in all font resources, and their details are listed below (see the OpenType Feature Registry for additional information):

- The 'halt' and 'vhal' GPOS feature are identical across all weights.

- The 'palt' and 'vpal' GPOS features for ExtraLight through Medium are identical, as are those for Bold and Heavy. These features cover the glyphs for kana, some full-width punctuation, some full-width symbols, full-width digits, and full-width Latin.
- The 'kern' GPOS feature includes weight-specific kerning pairs for proportional Latin, Greek, and Cyrillic glyphs, along with weight-independent kerning pairs for kana and some punctuation. The 'vkrn' GPOS feature includes only weight-independent kerning pairs for kana and some punctuation.

Changes

Version 1.002

Build Dates: March 18, 2015 & April 7, 2015. Built By: Dr. Ken Lunde (小林劍). Release Date: April 20, 2015.

General

- The open source license was changed from Apache License, Version 2.0 to SIL Open Font License, Version 1.1. This change is reflected in the *name.ID=13* (*License Description*) and *name.ID=14* (*License Info URL*) strings, and also in the stand-alone *LICENSE.txt* file.
- The copyright statement, as reflected in the *name.ID=0* (*Copyright notice*) strings, 'CFF ' tables, and CIDFont resources, was updated to reflect 2015 as the current year, and was adjusted in other ways.
- The OTFs and OTCs are additionally packaged as ZIP files that include all seven weights, for easier download from GitHub. See Issue #64.
- OTFs only for Regular and Bold, along with corresponding font instances in the OTCs and Super OTC, which differ from the existing ones only in that the default (encoded) glyphs for ASCII (U+0020 through U+007E), U+00A0 ( ), U+00A5 (¥), U+2011 (-), and U+20A9 (₩) are half-width instead of proportional were added. See Noto Issue #136.
- The 24 PUA mappings that are currently required for GB 18030 certification are now provided, for the benefit of developers who need to add them to the fonts' 'cmap' tables, in the *utf32-gb18030pua24.map* file.
- The *OS/2.achVendID* value was changed from ADBE to ADBO. See Issue #80.
- The *OS/2.sTypoLineGap*, *hhea.LineGap*, and *vhea.lineGap* values were changed from 500 to 0 (zero). The 7th bit of the *OS/2.fsSelection* value (*USE_TYPO_METRICS*) was also turned off. These changed are intended to provide more consistent cross-platform line spacing. See Issue #76, Issue #88, Noto Issue #74 & Noto Issue #126.
- The glyph for U+20AB (₫; *uni20AB*; CID+65485) was added. See Issue #91.
- Two Unicode mappings—for U+20AB (₫; *uni20AB*; CID+65485) and U+22EF (⋯; *uni2026*; CID+728)—were added to all CMap resources, which now include a total of 44,651 mappings.
- The mapping for U+39B3 (䦳) was changed to *uni363D-CN* (䦳; CID+3084), and the mapping for U+439B (䎛) was changed to *uni3588-CN* (䎛; CID+2877).
- UTF-32 CMap resources that differ from the existing ones only in that the default (encoded) glyphs for ASCII (U+0020 through U+007E), U+00A0 ( ), U+00A5 (¥), U+2011 (-), and U+20A9 (₩) are half-width instead of proportional were added.
- UTF-16 CMap resources were added for environments that require their use. The UTF-32 CMap resources should continue to be used for building the 'cmap' tables of OpenType/CFF fonts.
- CIDs 65485 through 65505 were added to all region-specific subset definitions.
- The glyph for U+2014 (—; *uni2014*; CID+715) is now handled via the 'locl' GSUB feature by making the glyph for U+2015 (―; *uni2015*; CID+716) its CJK form.
- The proportional glyphs for U+2014 (—; *uni2014*; CID+715), U+2026/U+22EF (…; *uni2026-PW*; CID+63038), U+2E3A (⸺; *uni2E3A*; CID+1346), and U+2E3B (⸻; *uni2E3B*; CID+1347) can now be substituted for their full-width (aka CJK) forms via the 'fwid' GSUB feature.
- The glyph for U+2015 (―; *uni2015*; CID+716), along with the full-width glyphs for U+2026/U+22EF (⋯; *uni2026*; CID+728), U+2E3A (⸺; *uni2E3A-JP*; CID+63028), and U+2E3B (⸻; *uni2E3B-JP*; CID+63029), can now be substituted for their proportional (aka Western) forms via the 'pwid' GSUB feature.

- The glyphs for U+FF3E (＾ ; *uniFF3E*; CID+59053) and U+FF40 (｀ ; *uniFF40*; CID+59055) now include 'vmtx' table overrides that lower them by 100 units, to keep them within the em-box in vertical writing. See Noto Issue #182.

Chinese (Simplified & Traditional)

- The mapping for U+2022 was changed to *uni2027* (‧ ; CID+729).

Simplified Chinese

- CIDs 4122 (*uni39B3-CN*) and 6950 (*uni439B-CN*) were removed from the CN region-specific subset definition.
- The following two Simplified Chinese glyphs were corrected by adding a short diagonal stroke to the top of the bottom component: *uni3862-CN* (幜; CID+3735) and *uni3B05-CN* (晜; CID+4499). See Issue #68.
- The Simplified Chinese glyph for U+5716 (圖; *uni5716-CN*; CID+13677) was adjusted by making its center vertical stroke perfectly vertical. See Issue #74.

Traditional Chinese

- The mapping for U+5716 was changed to *uni5716-JP* (圖; CID+13676). See Issue #74.
- CID+13678 (*uni5716-TW*) was removed from the TW region-specific subset definition, and CID+13676 (*uni5716-JP*) was added to it.
- The Traditional Chinese glyph for U+9014 (途; *uni9014-TW*; CID+40449) was adjusted by making its upper-right stroke curved outward instead of inward. See Issue #86.

Japanese

- The glyphs for <3042 3099> (あ゙; *uni3042uni3099*; CID+65486), <3044 3099> (い゙; *uni3044uni3099*; CID+65487), <3048 3099> (え゙; *uni3048uni3099*; CID+65488), <304A 3099> (お゙; *uni304Auni3099*; CID+65489), <3093 3099> (ん゙; *uni3093uni3099*; CID+65490), <30A2 3099> (ア゙; *uni30A2uni3099*; CID+65491), <30A4 3099> (イ゙; *uni30A4uni3099*; CID+65492), <30A8 3099> (エ゙; *uni30A8uni3099*; CID+65493), <30AA 3099> (オ゙; *uni30AAuni3099*; CID+65494), <30F3 3099> (ン゙; *uni30F3uni3099*; CID+65495), along with corresponding vertical forms (CIDs 65496 through 65505), were added, and are made accessible via the 'ccmp' GSUB feature.
- The following four Japanese glyphs were tweaked in very minor ways: *uni51F9-JP* (凹; CID+11517), *uni585A-JP* (塚; CID+14185), *uni5D07-JP* (崇; CID+16616), and *uni8868-JP* (表; CID+37004).
- The JIS X 0212 glyph for U+7C69 (籩; *uni7C69uE0101-JP*; CID+62483) was corrected by removing the short vertical stroke from its center whose presence made it identical to the default Japanese glyph (籩; *uni7C69-JP*; CID+30851).

Korean

- The glyphs for U+3131 through U+3163 and U+3165 through U+318E (*Hangul Compatibility Jamo*), along with those for U+FFA1 through U+FFBE, U+FFC2 through U+FFC7, U+FFCA through U+FFCF, U+FFD2 through U+FFD7, and U+FFDA through U+FFDC (*Halfwidth Hangul variants*), were mechanically centered along the Y-axis. See Noto Issue #266.
- The mapping for U+5173 was changed from *uni5173-JP* (关; CID+11297) to *uni5173-CN* (关; CID+11298), the mapping for U+590D was changed from *uni590D-JP* (复; CID+14484) to *uni590D-CN* (复; CID+14485), the mapping for U+7529 was changed from *uni7529-JP* (甩; CID+27454) to *uni7529-CN* (甩; CID+27455), and the mapping for U+95E8 was changed from *uni95E8-JP* (門; CID+43125) to *uni95E8-CN* (门; CID+43126). See Noto Issue #151.

Version 1.001

Build Date: September 3, 2014. Built By: Dr. Ken Lunde (小林劍). Release Date: September 12, 2014.

General

- The 65,535-glyph OTFs that are used to build the OTCs have been deployed as separate language-specific OTFs, mainly for user convenience.
- A Super OTC that includes all 28 language-specific OTFs was added as a fourth deployment format.
- The *IsBoldStyle* keyword value in the *cidfontinfo* files for the Bold and Heavy weights was changed from true to false. This change affects the *name.ID=2* and *name.ID=4* strings. See Issue #29.
- The *OS/2.fsType* (aka FSType) value was changed from 8 to 0 (zero). See Issue #58.
- The *OS/2.usWeightClass* value for the ExtraLight weight was changed from 100 to 250.
- The 'vert' GSUB feature substitutions for the following characters were removed: U+2016 (for better UTR #50 compliance), U+3099 (to unblock Unicode-based composition), and U+309A (to unblock Unicode-based composition). See Issue #60.
- The order of some of the per-language lookups in the 'locl' GSUB feature was changed.
- The proportional glyph for U+2026 (…; *uni2026-PW*; CID+63038) was adjusted so that the three periods that comprise it evenly distribute when strung together in multiples of two or more. This affected all weights except for Heavy. Note that the horizontal advances for this glyph changed as a result, from 1018 to 1101 units for ExtraLight, and from 1123 to 1144 units for Heavy. See Issue #26.
- The vertical glyphs for U+302A through U+302D (*uni302A-V*; CID+65160, *uni302B-V*; CID+65161, *uni302C-V*; CID+65162, and *uni302D-V*; CID+65163) continue to be zero-width in terms of vertical advance, but their horizontal advances were changed from 0 (zero) to 1000 units.

Chinese (Simplified & Traditional)

- The mapping for U+66F5 was changed from *uni66F5-JP* (曳; CID+21056) to *uni66F5uE0101-JP* (曳; CID+62265).

Simplified Chinese

- The Simplified Chinese glyphs for the following 12 characters were tweaked in minor ways: U+38D3 (㚔; *uni38D3-CN*; CID+3868), U+41C9 (䇉; *uni41C9-CN*; CID+6428), U+6248 (扈; *uni6248-CN*; CID+19026), U+6A9A (檚; *uni6A9A-CN*; CID+22668), U+6FC5 (濅; *uni6FC5-CN*; CID+24946), U+6FCB (濋; *uni6FCB-CN*; CID+24956), U+74B4 (璴; *uni74B4-CN*; CID+27214), U+790E (礎; *uni790E-CN*; CID+29130), U+85F4 (蘊; *uni85F4-CN*; CID+35893), U+9B2C (鬬; *uni9B2C-CN*; CID+45513), U+9BBA (鮝; *uni9BBA-CN*; CID+45755), and U+9F7C (齾; *uni9F7C-CN*; CID+47415).

Traditional Chinese

- The Traditional Chinese font resources were split into separate TW and HK font resources. This necessitated a change to the PostScript and Family names for the Traditional Chinese subset OTFs that now include only "TW" in their names. *For the benefit of those users who are not sensitive to the TW and HK glyph differences, the Traditional Chinese font resources actually remain unchanged, and other than a small number of corrections, the Traditional Chinese subset OTFs changed only in name.* The first HK-specific font resources will be released at a later date, will be considered experimental, will include all 65,535 glyphs, will repurpose existing glyphs to the extent that is possible, and will not be included in the OTCs. They shall be added to the OTCs when they are no longer considered experimental. See Issue #6, Issue #18 & Issue #48.
- The 'hist' (*Historical Forms*) GSUB feature was added, which substitutes the encoded version of U+3127 (ㄧ; *uni3127-V*; CID+65353) that is used for both writing directions by the historical horizontal form (丨; *uni3127*; CID+1676). Note that the 'vert' GSUB feature already provided a substitution from the historical horizontal form to the contemporary form that serves both writing directions.

- The glyphs for the following bopomofo characters were tweaked to improve their design: U+310D (ㄍ; *uni310D*; CID+1650), U+3122 (ㄢ; *uni3122*; CID+1671), U+3123 (ㄣ; *uni3123*; CID+1672), U+3125 (ㄥ; *uni3125*; CID+1674), and U+3127 (ㄧ; *uni3127-V*; CID+65353). See Issue #12.
- The Traditional Chinese glyph for U+FF0C (，; *uniFF0C-TW*; CID+63033) was too high, especially in the lighter weights. The glyphs for all weights were adjusted. See Issue #1.

Traditional Chinese (TW)

- The mapping for U+5605 was changed from *uni5605-JP* (嗰; CID+13198) to *uni5605-CN* (嗰; CID+13199). While this arguably makes the glyph more suitable for HK use, it results in greater consistency with other characters, and its Cantonese-specific use cases suggest that it is the better glyph for a broader set of users. See Issue #17 & Issue #55.
- The TW glyph for U+7921 (礡; *uni7921-TW*; CID+29163) was incorrect in that it included Radical #85 as a component, and was identical to the HK glyph for U+7934 (礡; *uni7934-HK*; CID+29200).
- The TW glyph for U+914D (配; *uni914D-TW*; CID+41090) was tweaked in a minor way.
- The TW glyph for U+9B54 (魔; *uni9B54-TW*; CID+45606) had its 鬼 component corrected. See Issue #23.

Japanese

- A small number of glyphs for kana were tweaked for improved readability.
- The alternate horizontal advances for U+30C1 (チ; *uni30C1*; CID+1579) and U+30C2 (ヂ; *uni30C2*; CID+1580), as specified in the 'palt' GPOS feature, were changed.
- The glyphs for U+3099 through U+309C (゙; *uni3099*; CID+1539, ゚; *uni309A*; CID+1540, ゙; *uni309B*; CID+1541, and ゚; *uni309C*; CID+1542), along with their vertical variants (〫; *uni3099-V*; CID+65252, 〬; *uni309A*-V; CID+65253, 〫; *uni309B-V*; CID+65254, and 〬; *uni309C-V*; CID+65255), were corrected so that they are in the correct position within the em-box. See Issue #59.
- The glyphs for U+3191 through U+319F (㆑; *uni3191*; CID+1777, ㆒; *uni3192*; CID+1778, ㆓; *uni3193*; CID+1779, ㆔; *uni3194*; CID+1780, ㆕; *uni3195*; CID+1781, ㆖; *uni3196*; CID+1782, ㆗; *uni3197*; CID+1783, ㆘; *uni3198*; CID+1784, ㆙; *uni3199*; CID+1785, ㆚; *uni319A*; CID+1786, ㆛; *uni319B*; CID+1787, ㆜; *uni319C*; CID+1788, ㆝; *uni319D*; CID+1789, ㆞; *uni319E*; CID+1790, and ㆟; *uni319F*; CID+1791) are no longer generic in terms of weight, and now differ by weight.
- The following eight Japanese glyphs were tweaked in minor ways, mainly related to their proportion within the em-box: *uni2E8C-JP* (丨; CID+1354), *uni2E8D-JP* (丶; CID+1355), *uni2EA9-JP* (王; CID+1364), *uni2ED7-JP* (雨; CID+1375), *uni793B-JP* (礻; CID+29211), *uni7F52-JP* (罒; CID+32249), *uni7F53-JP* (罓; CID+32251), and *uni7F8AuE0100-JP* (variant of 羊; CID+62526).
- The following two Japanese glyphs were adjusted to be left-justified in the em-box: *u29D4BuE0101-JP* (variant of 臭; CID+62992) and *uni8A01-JP* (言; CID+37771).
- The Japanese glyph for U+4E44 (乄; *uni4E44-JP*; CID+9928) had the vestigial serif removed from its upper-right corner.
- The Japanese glyph for U+510A (儊; *uni510A-JP*; CID+11108) was tweaked in a very minor way so that it also becomes suitable for Simplified Chinese use. As a result, its working glyph name was changed to *uni510A-CN* mainly for administrative purposes, but its CID and usage remain unchanged.
- The Japanese glyphs for U+617E (慾; *uni617E-JP*; CID+18660), U+7B8F (箏; *uni7B8F-JP*; CID+30378), U+7B98 (箘; *uni7B98-JP*; CID+30400), U+9A2B (騫; *uni9A2B-JP*; CID+45034), U+9AFA (髺; *uni9AFA-JP*; CID+45409), and U+9DD9 (鷙; *uni9DD9-JP*; CID+46597) had an issue with the Heavy master that affected all weights except for ExtraLight due to interpolation. See Issue #38.
- The Japanese glyph for U+791F (礟; *uni791F-JP*; CID+29158) had its middle component tweaked.

Korean

- Approximately 950 glyphs for hangul syllables were fine-tuned for better balance of their components.
- The lookups for the 'ljmo', 'vjmo', and 'tjmo' GSUB features were removed from the 'calt' GSUB feature, and the 'ljmo', 'vjmo', and 'tjmo' GSUB features were changed to declare only the 'hang' script, along with the 'KOR' and 'dflt' languages. See Issue #47.
- The first of two combining forms of U+1198 (ᆘ; *uni1198.vjmo01*; CID+64463) and U+1199 (ᆙ; *uni1199.vjmo01*; CID+64464) were added to the backtrack for the 'tjmo_third' lookup. See Issue #9.
- The nominal (encoded) glyphs for U+119F (ᆟ; *uni119F*; CID+524) and U+11A1 (ᆡ; *uni11A1*; CID+526) were incorrect in that their vertical bar should have extend more downward. The nominal forms of U+119D (ᆝ ; *uni119D*; CID+522) and U+11A0 (ᆠ; *uni11A0*; CID+525) were also similarly tweaked. See Issue #33.
- The second of two combining forms of U+119F (ᆟ; *uni119F.vjmo02*; CID+64565) was incorrect in that its vertical bar was too short and should have extend downward, similar to that of the corresponding combining form of U+11A1 (ᆡ; *uni11A1.vjmo02*; CID+64567), which was also tweaked. See Issue #33.
- The nominal (encoded) glyph for U+11A0 (ᆠ; *uni11A0*; CID+525) was added to the first lookahead for the 'ljmo_second' lookup and to the only lookahead for the 'ljmo_fifth' lookup. See Issue #9.
- The vertical glyphs for U+302E and U+302F (*uni302E-V*; CID+65164 and *uni302F-V*; CID+65165) required an adjustment whereby their positions are as intended when they are swapped with the character that they proceed, which required a change to their vertical origins. They continue to be zero-width in terms of vertical advance, but their horizontal advances were changed to 920 units. See Issue #34.
- The glyph for U+3273 (㉳; *uni3273*; CID+2005) was missing an inner subpath in the Heavy weight.
- The fifth of six combining forms of U+A966 (ꥦ; *uniA966.ljmo05*; CID+64259) had an incorrect right-side component.
- The glyphs for U+B090 (낐; *uniB090*; CID+48734) and U+BE64 (빤; *uniBE64*; CID+52274) had a connection issue in the Heavy master, which affected all weights except for ExtraLight.
- The glyphs for U+B0DF (낟; *uniB0DF*; CID+48813), U+D177 (텷; *uniD177*; CID+57157), U+D713 (훓; *uniD713*; CID+58593), and U+D717 (훗; *uniD717*; CID+58597) were corrected to use a vertical stroke for the top stroke of the ㅎ component. See Issue #22.
- The glyphs for U+B84E (롎; *uniB84E*; CID+50716) and U+D07A (큺; *uniD07A*; CID+56904) had a connection issue that affected all weights.
- The glyph for U+C974 (쥴; *uniC974*; CID+55106) had a connection issue in the ExtraLight master, which affected all weights except for Heavy.
- The nominal (encoded) glyph for U+D7B0 (ힰ; *uniD7B0*; CID+58738) was incorrect in that it was identical to the nominal glyph for U+117D (ᅽ; *uni117D*; CID+490). See Issue #9.
- The first of two combining forms of U+D7B0 (ힰ; *uniD7B0.vjmo01*; CID+64479) was added to the backtrack for the 'tjmo_second' lookup. See Issue #9.

Version 1.000

Build Date: June 24, 2014. Built By: Dr. Ken Lunde (小林劍). Release Date: July 15, 2014.

First public release.

Known Issues

Please report all issues in the GitHub repository so that they can be properly tracked, and for greater visibility among the user community. Also, be sure to thoroughly check the closed issues prior to submitting a new issue.

Because these fonts exercise several architectural limits, particularly the ones that include 65,535 glyphs, some environments may have difficulties using them properly, sometimes due to implementation limits or poor assumptions. If this is the case, please report such issues so that they can be recorded and tracked. You are also strongly encouraged to contact the developer of such environments to report the same.

General

Various component consistency issues, glyph corrections, and a greater degree of glyph sharing—particularly between Japanese and Simplified Chinese—are planned to be reflected in the next major version. See Issue #93, Issue #96, & Issue #97. **IMPORTANT NOTE**: suggestions for glyph sharing and glyph corrections shall be consolidated in Issue #98 and Issue #99, respectively, which should also be used to report all future suggestions.

Traditional Chinese

- The possibility of improving the shape of the 辶 (U+8FB6) component is still under investigation. See Issue #50.

Traditional Chinese (TW)

- Glyphs that use 又 (U+53C8) or 叉 (U+53C9) as components, including these glyphs themselves, are still being checked for correctness according to the Taiwan MOE glyph standard. The former is correct (closed), but the latter is not, and should instead be open, like the JP form (叉; *uni53C9-JP*; CID+12279). See Issue #61.

Traditional Chinese (HK)

- Experimental HK fonts still need to be created, which is to be done in phases. The first phase, completed as of Version 1.001, was to separate Traditional Chinese into separate TW and HK font resources. The second phase is to develop experimental HK fonts that repurpose existing glyphs to the extent that is possible. See Issue #6, Issue #18, Issue #48, Issue #89, Issue #92, Noto Issue #42 & Noto Issue #70.
- The HK glyph for U+7740 (着; *uni7740-CN*; CID+28385) is inappropriate, and the UTF-32 CMap resource (and the corresponding 'cmap' tables) for HK should instead map this code point to the JP glyph (着; *uni7740-JP*; CID+28384). This change is not reflected in this update because it is an HK-specific issue. See Issue #23.

Korean

- The glyphs for *uni11C1.tjmo04* (CID+65033) and *uni11D6.tjmo04* (CID+65054) need to be shifted to the right in order to better align with other jamo when combined. This issue is manifested in the Heavy weight, but affects all other weights except for ExtraLight due to interpolation. See Issue #90.
- Combining jamo doesn't work in vertical writing mode. See Issue #34.
- Consider adding glyphs for non–full-width versions of bracket-like characters that lack an encoded non–full-width counterpart: U+3008 & U+3009 (〈〉), U+300A & U+300B (《》), U+300C & U+300D (「」), U+300E & U+300F (『』), U+3010 & U+3011 (【】), U+3014 & U+3015 (〔〕), U+3016 & U+3017 (〖〗), U+3018 & U+3019 (〘〙), U+301A & U+301B (〚〛), and U+FF5F U+FF60 (｟｠). See Noto Issue #120.

That is all.