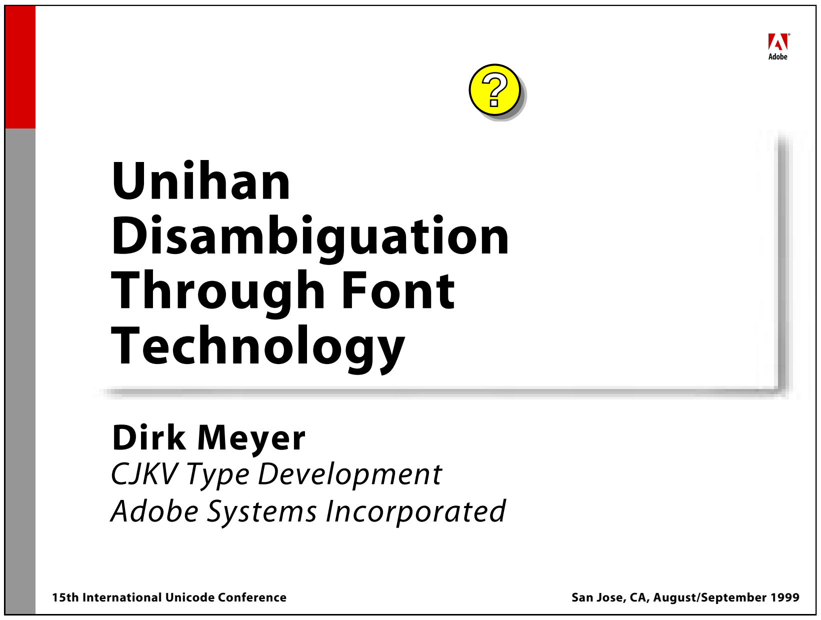
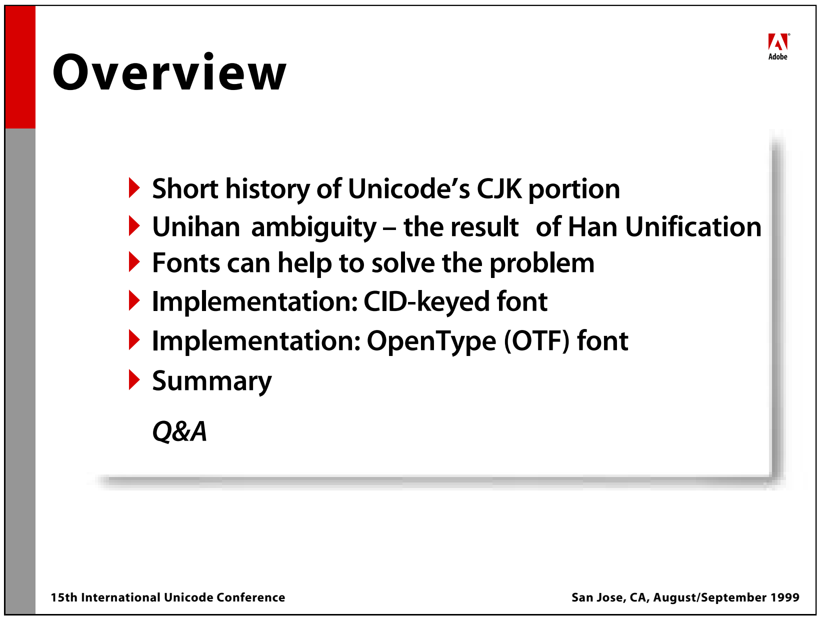


Unihan Disambiguation Through Font Technology

Dirk Meyer
CJKV Type Development
Adobe Systems Incorporated

15th International Unicode Conference San Jose, CA, August/September 1999



Overview

- Short history of Unicode's CJK portion
- Unihan ambiguity – the result of Han Unification
- Fonts can help to solve the problem
- Implementation: CID-keyed font
- Implementation: OpenType (OTF) font
- Summary

Q&A

15th International Unicode Conference San Jose, CA, August/September 1999

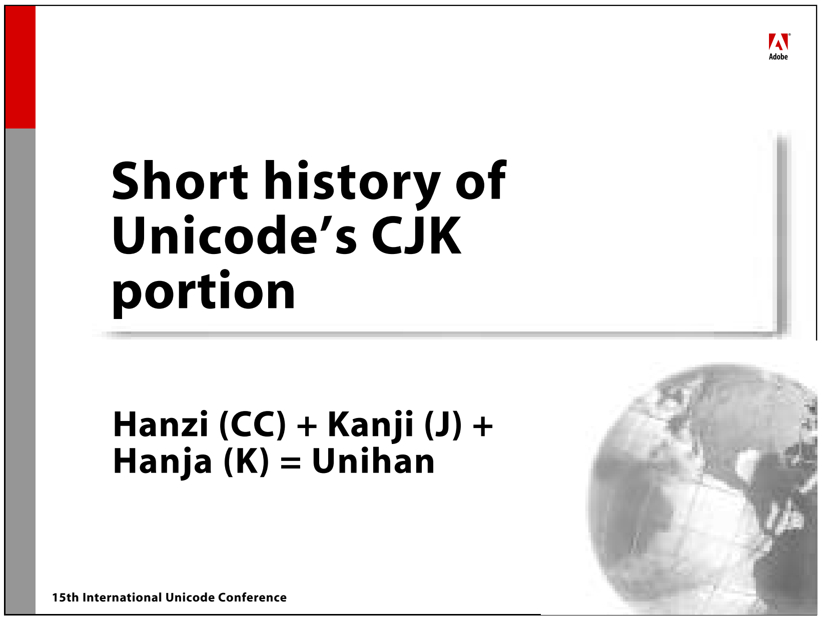
"Unihan disambiguation" Through Font Technology

The purpose of this presentation is to show how different font technologies (CID-keyed Font Technology, OpenType, etc.) can be applied to help resolving what is commonly called the "Unihan ambiguity problem."

The process of Han Unification can be considered to be one of the major "historical" achievements among the efforts to create Unicode. But developers are facing the problem of how to "disambiguate" the characters of the Basic Multilingual Plane's (BMP) Unihan portion **in the context of cross-locale Unicode fonts**.

In order to represent the Chinese characters of different Asian locales in a culturally adequate and typographically correct way with the help of Unicode, additional glyphs must be available in a font which shall be used across locale borders. Preliminary research shows that in such a "multi-locale" or "Pan-CJK" font, roughly 50 percent of the CJK characters need more than one glyph representation, depending on the typeface.

Different approaches exist to make the additional glyphs available in fonts and how applications can get access to them. This presentation will provide implementation examples for achieving it through fonts applying CID-keyed or the closely related OpenType font technology. It will focus on explaining and demonstrating how the problematic consequences of Han Unification can be resolved with the help of fonts.



Short history of Unicode’s CJK portion

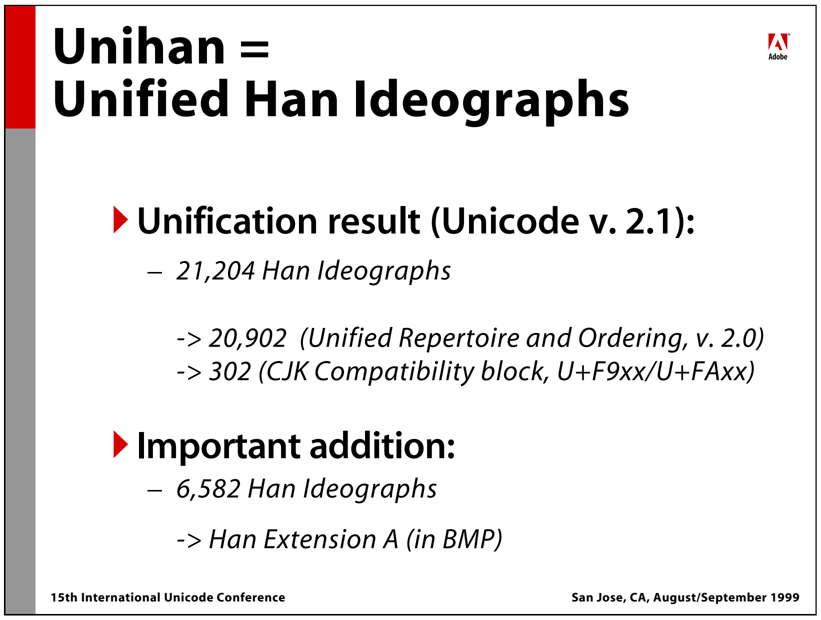
Hanzi (CC) + Kanji (J) + Hanja (K) = Unihan

15th International Unicode Conference

In the process of defining Unicode, the Han Unification is probably the biggest achievement overall. Before the creation of Unicode, several Asian countries had established encoded or unencoded Han character sets with partly overlapping contents. In order to make the “unified repertoire of Han ideographs” a reality, representatives of these countries put a lot of joint effort into phrasing precise rules about how to treat, in a common code, those characters that were different, or “nearly different.”

Basically, these rules define which characters from different locales can – despite their sometimes-subtle differences – be considered identical (and thus be unified in order to occupy a single code point), and which are too different to be unified. To make it completely clear: the differences referred to here are not, for example, those between traditional and simplified Chinese characters. Han Unification takes place where the same character is written differently in Japanese or Chinese, because different typographical or glyph design rules exist in these countries.

Han Unification freed up many otherwise wasted code points and helped to avoid duplicately-encoded characters. Of course, exceptions exist, but the procedures made and still make a lot of sense.



Unihan = Unified Han Ideographs

- **Unification result (Unicode v. 2.1):**
  - *21,204 Han Ideographs*

    -> 20,902 (Unified Repertoire and Ordering, v. 2.0)
    -> 302 (CJK Compatibility block, U+F9xx/U+FAxx)
- **Important addition:**
  - *6,582 Han Ideographs*

    -> Han Extension A (in BMP)

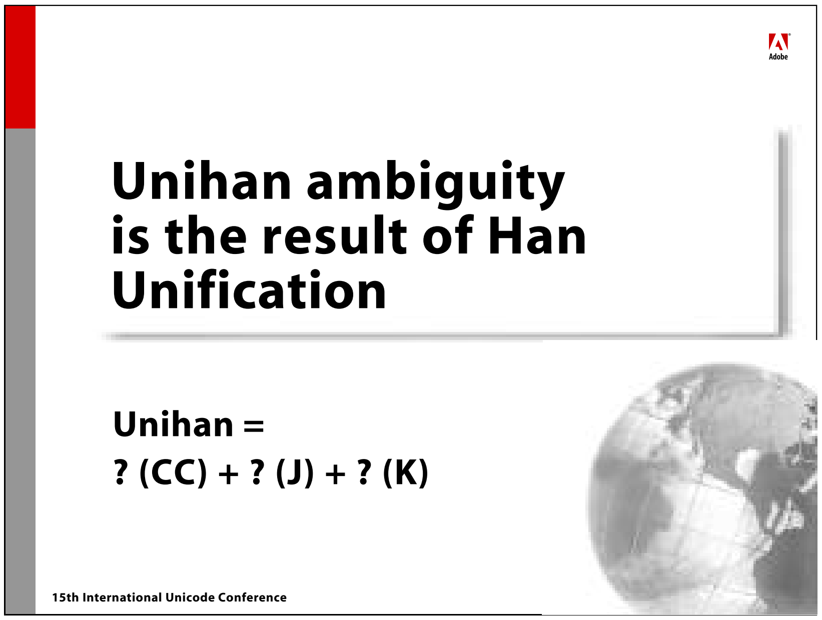
15th International Unicode Conference — San Jose, CA, August/September 1999

Based on those Han Unification rules, the process of extending Unified Repertoire and Ordering (URO) both "horizontally" (to include **character mappings** from other or newly established standards [Hong Kong SAR, Vietnam]) and "vertically" (to include **new characters** from these standards) is continuing and will continue for years to come.

For additional information about the Han Unification process, see: *Han Unification History [Appendix E,]* *(The Unicode Standard, Version 2.0, pp. E-1f)*

For information about Unicode's CJK source standards, structure and ordering of Unihan, as well as exceptions for the Han Unification process (like the "source separation rule", "non-cognate rule"), see: *CJK Unified Ideographs: U+4E00–U+9FFF [CJK Ideographs Area,]* *(The Unicode Standard, Version 2.0, p. 6-104ff)*

For explanations about the source properties of each Unihan character, see: *CJK Unified Ideographs [Code Charts, Chapter 7.2]* *(The Unicode Standard, Version 2.0, p. 7-3f)*



Unihan ambiguity is the result of Han Unification

Unihan =

? (CC) + ? (J) + ? (K)

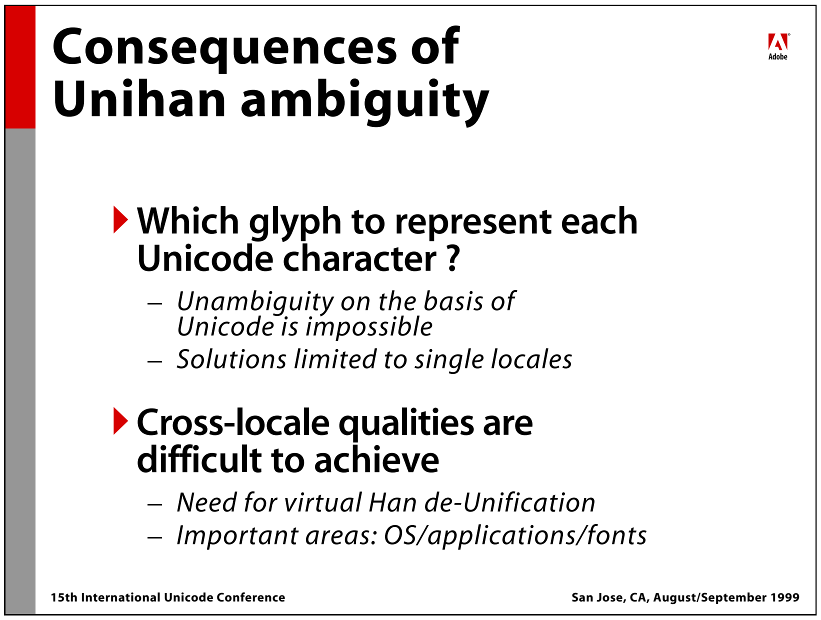
15th International Unicode Conference

Not always fully understood are the consequences rooting in the fact that Unicode is a “character” standard and thus does not define any character shapes or “glyphs.” In other words, it does not care about specific representations of given (“abstract”) characters. Only this precondition made a process like the one of Han Unification possible in the first place. However, we now must face the problem of Unihan ambiguity as its direct outcome: “Welcome to the artificial world of Unihan ideographs.”

In other words, characters – represented differently throughout different Asian locales – have been unified into a single Unicode code point. How is it possible for a user or an application of a certain locale to get back to the origin – the correct glyph when using Unicode?

If the target destination for an operating system, an application, or a font is only one single locale, it is sufficient to use one glyph to represent a Unicode CJK code point.

Problems occur in a multi-locale context: in order to again get the “original,” often differing locale-specific glyphs, the Han unification process has to be reversed. During this reverted process, however, no information is provided about any glyph differences when a Unicode character is rendered in (or for the use in) different locales. Any information about them has to be kept at different locations, for example, in fonts.



Consequences of Unihan ambiguity

- **Which glyph to represent each Unicode character ?**
  - *Unambiguity on the basis of Unicode is impossible*
  - *Solutions limited to single locales*
- **Cross-locale qualities are difficult to achieve**
  - *Need for virtual Han de-Unification*
  - *Important areas: OS/applications/fonts*

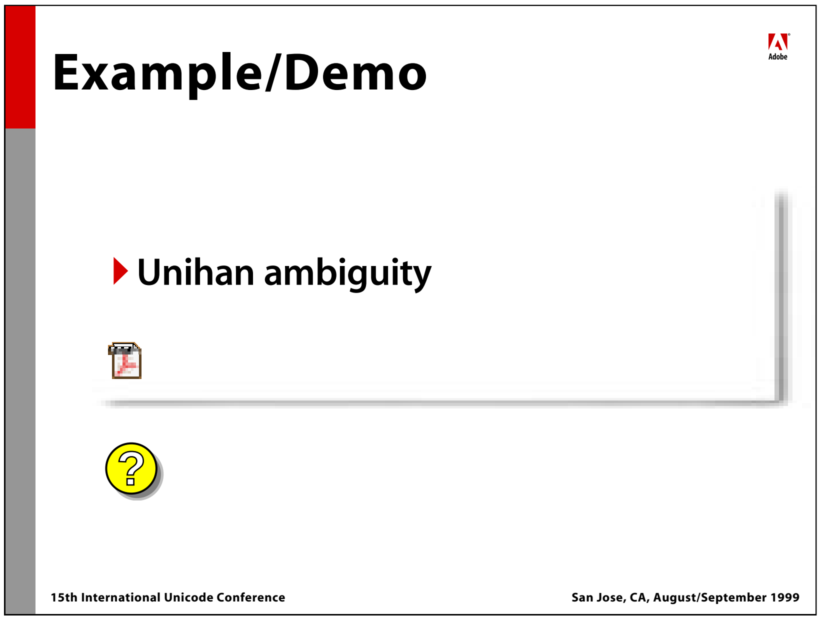
15th International Unicode Conference San Jose, CA, August/September 1999

Sometimes is does not matter, sometimes it does: the inherent logic of Han Unification implies that it is impossible to work on the basis of Unicode, and – at the same time – achieve Unicode CJK output that is equally accepted throughout all CJK locales.

If there has been a Han Unification to create a common character set (Unihan), it takes a virtual “de-Unification” whenever unambiguity is needed. This is true for the visual output of all Han ideographs affected by Han unification.

No matter what a “Unicode product” claims to be (or is taken for by its users), anything based on the principle of “one CJK glyph per character code” can only serve the needs of a single locale. It is limited in its use to a single locale, because a user cannot rely on complete accuracy or typographical correctness for all glyphs when it comes to cross-locale usage.

Obviously, localized versions of an operating system, applications or fonts that are intended to be used in one CJK locale only do not need correct glyphs for each locale because only the “native” one is of concern. It is, however, fairly easy to imagine situations in which operating systems or applications that have “locale bridging” character might benefit from a mechanism which is able to serve more than one locale.



Example/Demo

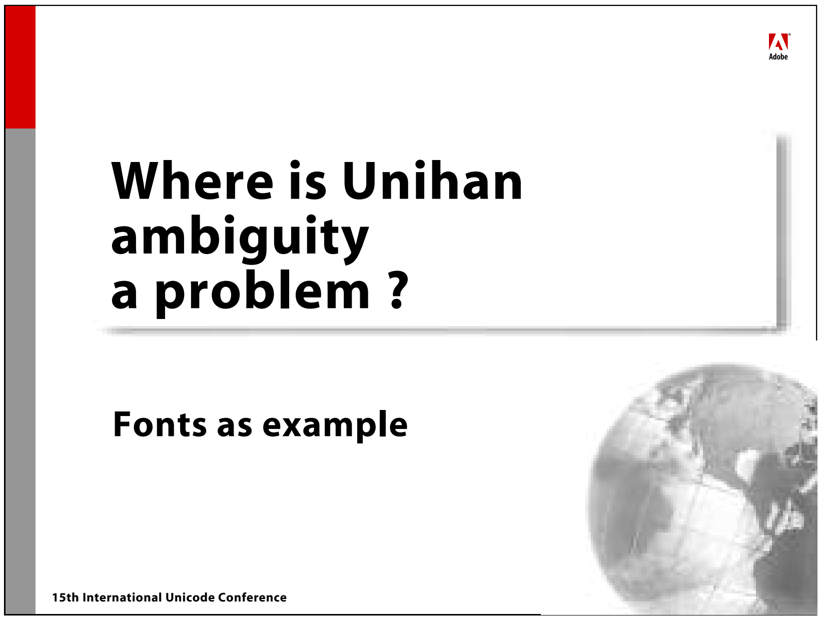
- Unihan ambiguity

15th International Unicode Conference San Jose, CA, August/September 1999

Not very many Unicode characters have four different representations: one for each of the four Asian locales CS, CT, J, and K. The majority have two or three different ones. The Acrobat PDF file shows examples for several characters having four variants. These examples illustrate how subtle the glyph differences across locales can be (areas that show modifications are indicated by shaded circles).

Note, how – according to the rules of Han Unification – one Unicode code point (U+) is used to represent four valid glyphs from four locales (G – T – J – K). "Han de-Unification" is necessary when correct glyphs for more than one locale are needed, for example, in a font.

[Note: A printout of the sample referred to on this slide will be handed out prior to the presentation.]



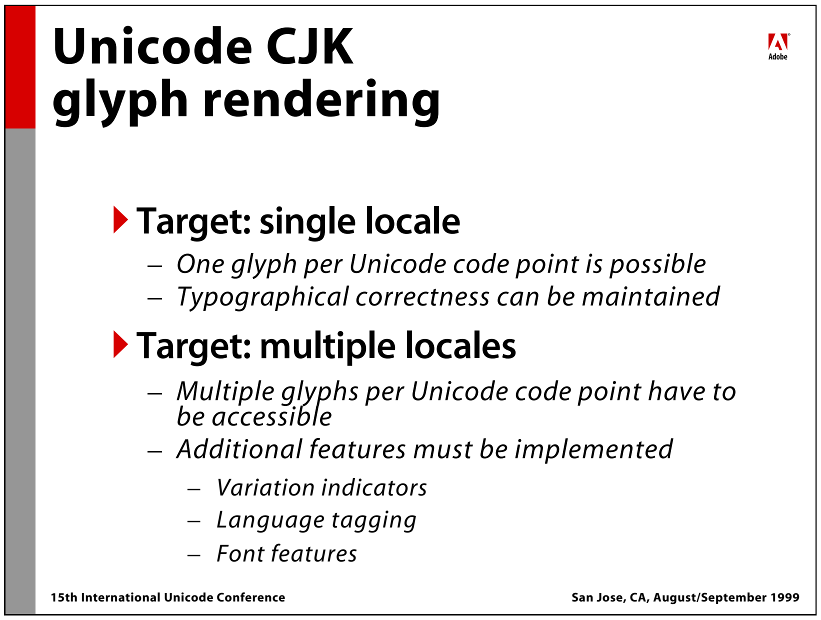
Where is Unihan ambiguity a problem ?

Fonts as example

15th International Unicode Conference

Parallel to the growing popularity of Unicode-based operating systems, a special kind of font product enjoys the increasing sympathy of more and more users in the “CJK arena”: Unicode fonts. [In the context of this presentation, the term in used to describe the intention to fully cover at least Unicode’s CJK character portion.] Such fonts seem to promise unlimited access to all collected Han ideographs and thus the capability to create texts in all languages based on these ideographs. Is this really true?

In general, the number of glyphs inside a given font can differ significantly depending on the target locale(s). Including the Han ideographs, Unicode provides roughly 40,000 characters, and a huge expansion can be expected from the advent of Unicode Version 3.0. Fonts that carry such a big repertoire of characters will create a huge overload in environments where scripts with alphabetic properties are used exclusively. There, Unicode fonts are rare, because it seems sufficient if a font carries one alphabet and perhaps parts of or a complete additional one. This assumption is supported by another obvious advantage of alphabetic scripts: in many cases, like Roman, Greek or Cyrillic, these scripts can easily be extended to represent – sometimes completely – more than one language in one font file.



Unicode CJK glyph rendering

- **Target: single locale**
  - *One glyph per Unicode code point is possible*
  - *Typographical correctness can be maintained*
- **Target: multiple locales**
  - *Multiple glyphs per Unicode code point have to be accessible*
  - *Additional features must be implemented*
    - *Variation indicators*
    - *Language tagging*
    - *Font features*

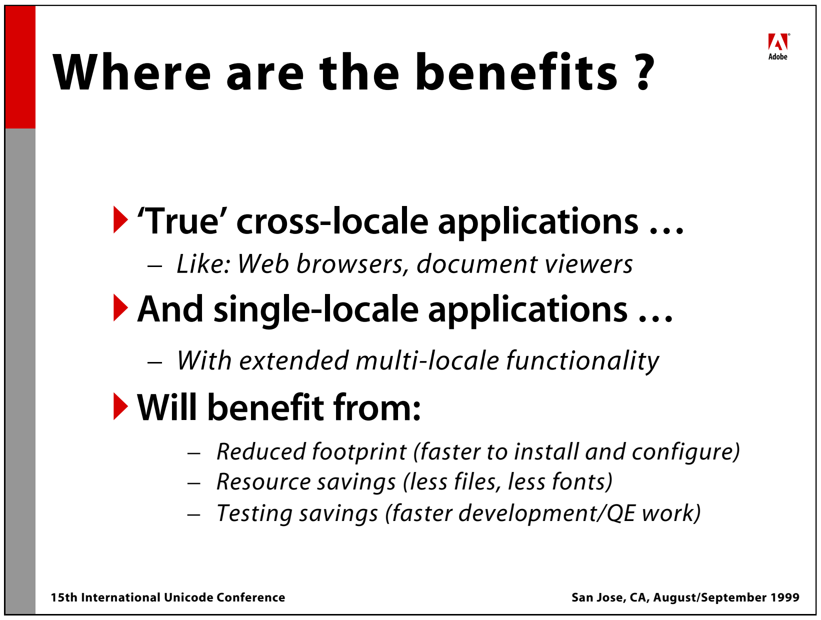
15th International Unicode Conference San Jose, CA, August/September 1999

As soon as we enter Unicode's "ideographic world" (U+4E00–U+9FA5), however, the situation changes, due to the results of Han Unification. First of all, we can somewhat naturally define two different kinds of Unicode CJK-fonts, depending on the number of locales they want to serve.

Minimum requirement for a **single-locale Unicode CJK-font** is, of course, the complete glyph coverage of its target locale. Users in that locale will only be satisfied if such a font allows them to have access to their locale-specific portion of Han ideographs, no matter whether they are called hanzi, kanji, or hanja. As long as a font contains the typographically correct glyphs, it is of minor importance, whether it covers the complete CJK Unihan range or includes only those glyphs actually used in the specific locale: user acceptance (in a single locale) is very likely.

The other "flavor" of such fonts, a **multiple-locale Unicode CJK-font** tries to cross borders and aims at serving more than one of the CJK locales. In this case, it is forced to "reversely apply" the rules of Han Unification and supply more than only one glyph whenever different representations of Unicode characters exist for all the fonts' destinations. There exist reasons for and against such fonts. Additional research and design efforts are necessary for their development.

In any case: no matter, whether a "system font" or an advanced font for high-end publishing purposes tries to be a "Pan-CJK font", they all must implement mechanisms to achieve a virtual "Han de-Unification."



Where are the benefits ?

- 'True' cross-locale applications …
  - *Like: Web browsers, document viewers*
- And single-locale applications …
  - *With extended multi-locale functionality*
- Will benefit from:
  - *Reduced footprint (faster to install and configure)*
  - *Resource savings (less files, less fonts)*
  - *Testing savings (faster development/QE work)*

15th International Unicode Conference San Jose, CA, August/September 1999

Developing a cross-locale font requires a considerable amount of research and effort. This brings up the question, who or what would benefit from Unihan disambiguation or correct cross-locale CJK functionality?

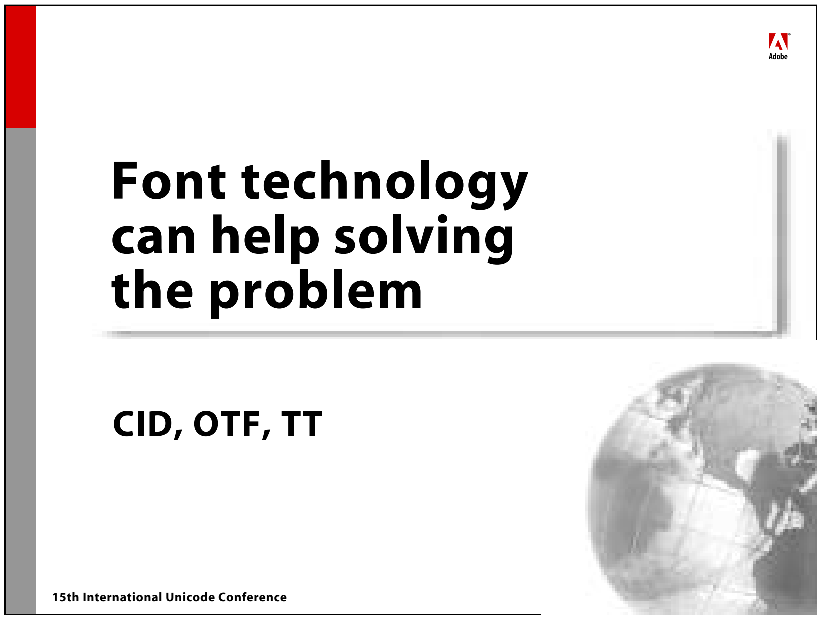
There are quite a few areas in which implemented cross-locale capabilities would serve both developers and users.

Some examples:

– Both Web browsers and document viewers could easily render incoming multi-lingual data stream correctly with only font installed and configured instead of three or four.

– From a user's point of view: text processing, publishing, and layout software could use a single font (thus a single typeface) in a single configuration to correctly create documents that are destined for different locales; no need to change fonts, switch to another localized version of the same application or even to another localized system.

– From a developer's point of view: a reduced number of fonts would considerably reduce development, testing and quality engineering work for installation, configuration, and application functionality.



Font technology can help solving the problem

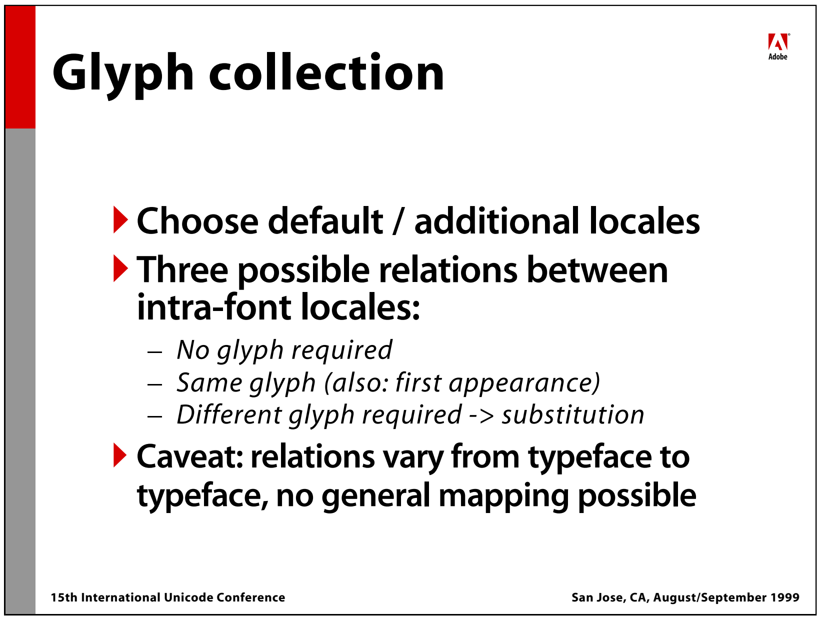
CID, OTF, TT

15th International Unicode Conference

Font technology currently provides at least three different approaches that allow for more than one glyph per code point. At the same time, these technologies make it possible to avoid duplication of characters inside a font file. Both properties are important to realize an economic, multi-locale Unicode CJK-font.

A Unicode-based “multi-locale” or “Pan-CJK” font can be created applying CID-keyed, OpenType (OTF), or TrueType (TT) font technology. This presentation will focus on examples of CID-keyed and OpenType technology.

The examples presented here describe the attempt to create a font that provides the correct glyphs for four Asian locales: Simplified Chinese, Traditional Chinese, Japanese, and Korean.



Glyph collection

- Choose default / additional locales
- Three possible relations between intra-font locales:
  - *No glyph required*
  - *Same glyph (also: first appearance)*
  - *Different glyph required -> substitution*
- Caveat: relations vary from typeface to typeface, no general mapping possible

15th International Unicode Conference

San Jose, CA, August/September 1999

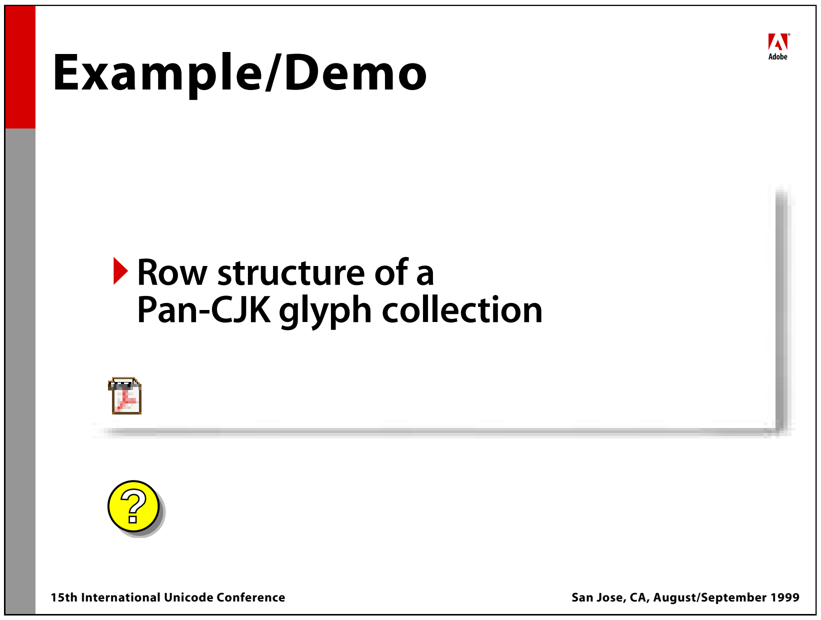
The development of the Pan-CJK prototype in both CID-keyed and OpenType flavors was based on outline data coming from a type foundry in the People's Republic of China. Consequently, the glyphs designed according to the rules of that locale were taken as the default.

A decision about a default locale is important. It sets the default design for all glyphs and, at the same time, it defines the number of additional glyphs that have to be designed for the other locales also covered by the font:

– If, at the same code point, no glyph exists in the additional locale no substitution needs to take place, and the default glyph can be used instead (to represent the full Unihan character repertoire, for example). [One could decide to design all glyphs as if they were used in all locales, even if they do not exist there, but designing non-existing, "artificial characters" does not seem to make much sense.]

– If the same glyph is required for an additional locale, again no substitution has to take place, one glyph can be used for two or more locales. Such a mechanism can also be applied in cases where a glyph is exclusively used outside the default locale: then it is designed in its "native" style, but serves as the default glyph.

– If the additional locale requires a different glyph, then another code-to-glyph mapping has to be activated or a glyph substitution mechanism has to be invoked.



Example/Demo

- Row structure of a Pan-CJK glyph collection

15th International Unicode Conference San Jose, CA, August/September 1999

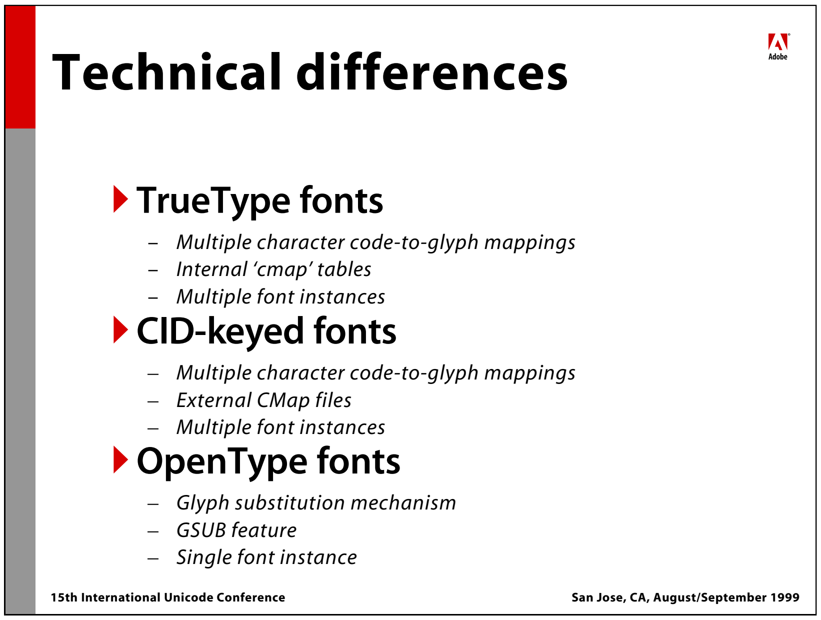
The example shows in which way the glyphs necessary for the support of multiple locales are collected and ordered. The G-locale is taken as the default, all glyphs are designed according to the rules of G. If a different glyph for the same code point has to be available for another supported locale, it has been added right after the default glyph.

This structure does not yet show, however, where the glyphs are actually used. Certain glyphs may not be required in the default locale at all, others will be re-used for additional locales.

The typographic style (the typeface) that is used for the design of the font plays an important role when it comes to deciding how many glyphs have to be available to cover the locales. Common styles for Chinese ideographs are Hei, Song, Fangsong, or Kai, for example. Depending on the style and its specific rules of how glyphs are composed, one or more parts of a character, combinations of strokes, or the connections between strokes, can be different among typefaces. In addition, different typeface-specific rules may exist in the target locales. These two levels of possible variations have significant influence on the internal structure of Pan-CJK character collections.

In other words, even if, according to Song-style design rules, a glyph is different in the Japanese and the Korean locale, the same glyph may be identical in both locales when designed for a Hei-style font. Accordingly, new differences may appear in a Fangsong-style font between locale-specific glyphs that were identical in a Kai-style font.

[Note: A printout of the sample referred to on this slide will be handed out prior to the presentation.]



Technical differences

- **TrueType fonts**
  - *Multiple character code-to-glyph mappings*
  - *Internal 'cmap' tables*
  - *Multiple font instances*
- **CID-keyed fonts**
  - *Multiple character code-to-glyph mappings*
  - *External CMap files*
  - *Multiple font instances*
- **OpenType fonts**
  - *Glyph substitution mechanism*
  - *GSUB feature*
  - *Single font instance*

15th International Unicode Conference San Jose, CA, August/September 1999

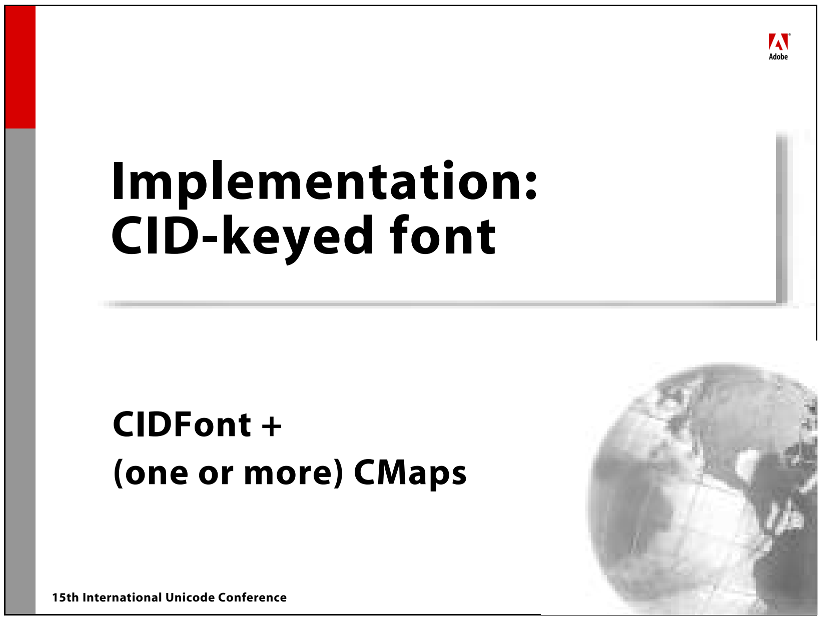
The TrueType font specifications include the option to provide multiple character to glyph mappings in a font.

For purposes as described here, a TT font file would contain the set of CJK glyphs intended to serve some or all locales without glyph duplication and more than one 'cmap' table inside the file. These 'cmap' tables establish a link between the character encoding and the glyphs contained inside the font. It is important to mention that glyphs can be referenced by different 'cmap' tables. Depending on the number of tables, a TT font file can offer different font names to the 'outside world': operating system and applications. Thus, it is a potential candidate to create a valid Pan-CJK font file in that it offers multiple "virtual" fonts, which when combined provide multi-locale functionality.

The two font formats CID-keyed and OpenType use a format different from TT to describe the glyph outlines in the font file.

Besides that, CID-keyed font technology offers a functionality similar to TT by using different external CMap files to access the glyphs in a second CIDFont file.

OpenType fonts may contain outline descriptions in both TrueType and Type-1 format. In addition to that, they provide new glyph substitution (GSUB) and glyph positioning (GPOS) mechanisms. Glyph substitution can effectively be used to achieve Pan-CJK functionality.



Implementation: CID-keyed font

CIDFont +
(one or more) CMaps

15th International Unicode Conference

CID-keyed font technology was especially designed to handle large numbers of glyphs in a single font file. This technology is currently gaining more and more market share in Asia, because it perfectly fits the needs of users there. However, nothing prevents this technology from being used for larger Latin-based character collections, too.

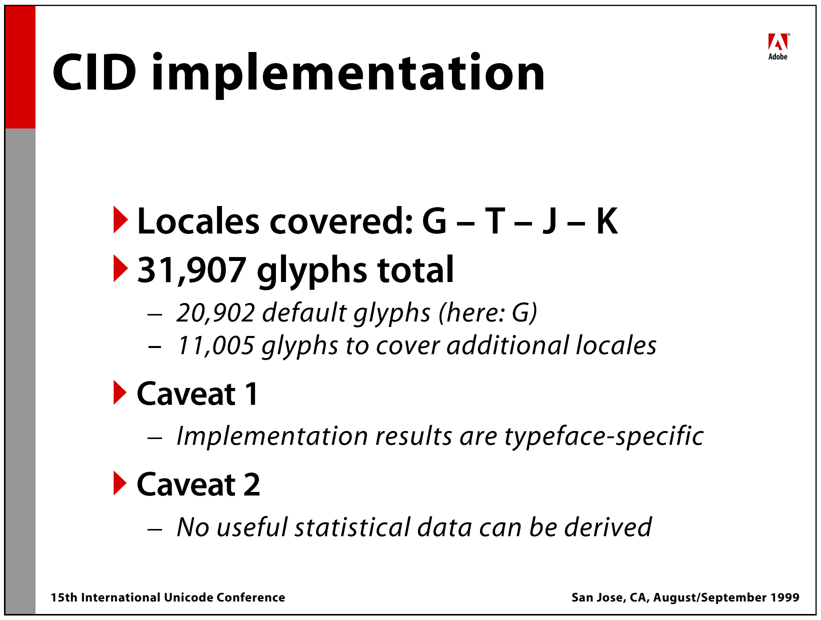
The technology is based on the interaction of two different file types:

– A CIDFont file contains only the outline descriptions of the glyphs, which are numbered in sequence according to their CID (character identifier) value starting from 0;

– A CMap file (not to be mixed up with the ‘cmap’ tables mentioned before) is a small entity separate from the CIDFont file. It maps character codes to glyphs inside the CIDFont file.

The combination of a CIDFont file and a CMap file creates a specific font instance, in which the glyphs inside the font file are mapped to whatever encoding is specified inside the CMap file.

Thus, it only takes a different CMap file to “repurpose” or “re-encode” the contents of the CIDFont file.



CID implementation

- Locales covered: G – T – J – K
- 31,907 glyphs total
  - *20,902 default glyphs (here: G)*
  - *11,005 glyphs to cover additional locales*
- Caveat 1
  - *Implementation results are typeface-specific*
- Caveat 2
  - *No useful statistical data can be derived*

15th International Unicode Conference San Jose, CA, August/September 1999

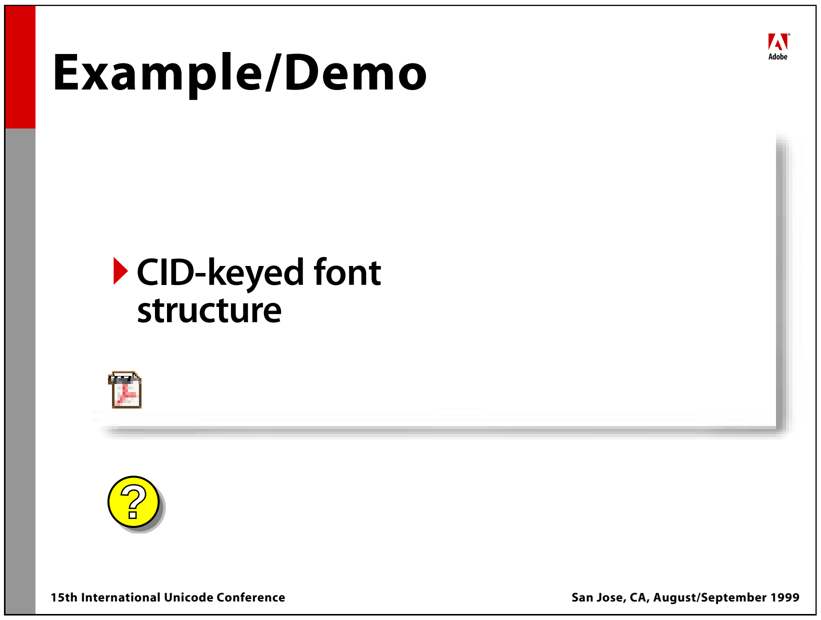
In our study, the Pan-CJK font is based on a Song design. 11,005 glyphs had to be added to the 20,902 glyphs representing the Unicode Unihan character portion in order to achieve typographical correctness and acceptable cross-locale results.

The locales covered by the fonts' glyph repertoire are that of Simplified and Traditional Chinese, Japanese, and Korean.

Again, it has to be kept in mind that the total number of 33,907 glyphs contained in this example represents an approach for a Song-style typeface. The glyph count will differ for other typefaces used to design Chinese ideographs.

Also, the number of 11,005 glyphs does not at all imply that this is the number of glyph differences between the default and the other locales. Sometimes a glyph is used in all locales, sometimes in only one. In the same way, an identical form may be used in locale A and C, while for locale B a special form exists, or no form at all.

And again, all this differs from design to design, and no especially useful statistical data can be derived from these differences.



Example/Demo

- CID-keyed font structure

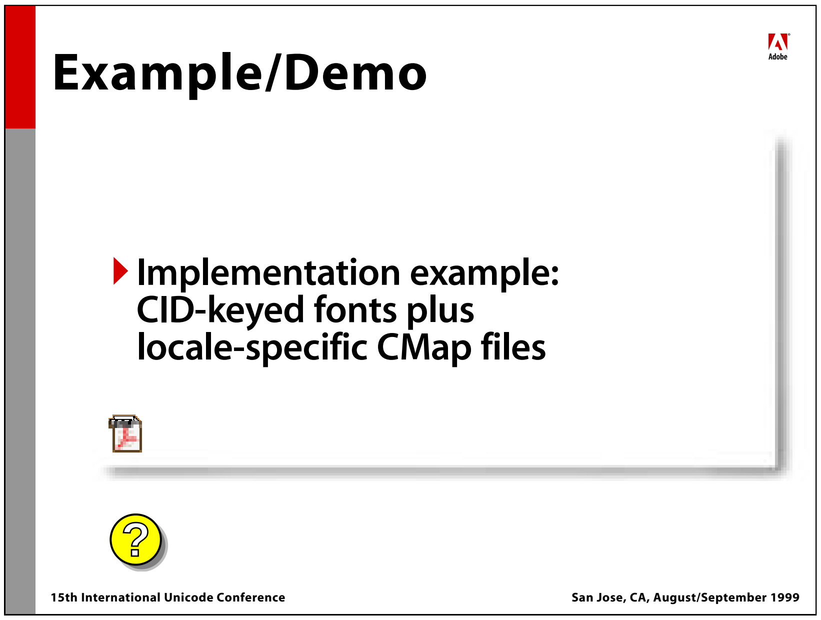
15th International Unicode Conference San Jose, CA, August/September 1999

This example shows how the internal font structure places the default glyphs first and adds additional locale-specific glyphs where necessary. This internal glyph ordering structure is common for both CID-keyed and OpenType font technologies.

In the case of the CID implementation example, all glyphs of the Pan-CJK font are contained in the CIDFont file and numbered in sequence from 1[one] through 33907 (CID 0[zero] remains reserved as the “undefined” glyph, which is used whenever no glyph is available for a certain code point).

The specific examples show characters which have up to four different glyph representations, one for each of the font’s target locales.

[Note: A printout of the sample referred to on this slide will be handed out prior to the presentation.]



Example/Demo

- Implementation example: CID-keyed fonts plus locale-specific CMap files

15th International Unicode Conference

San Jose, CA, August/September 1999

In CID-keyed font technology, CIDFont files form valid font instances only in combination with external CMap files. On a hard disk attached to a printer supporting Postscript (Version 2015 and higher), for example, the CIDFont file 'STSongCJK-Light' and the CMap file 'koKR-UCS2-H' create the font instance 'STSongCJK-Light--koKR-UCS2-H'. Based on a UCS2 encoding, this instance provides the Unihan glyphs according to Song typeface design-rules as they are written in the Korean locale for horizontal writing direction.

In order to build the complete multi-locale CID solution four different CMap files were created to use and re-use the character repertoire in the CIDFont file for four different locales :

– 'zhCN-UCS2-H' (Simplified Chinese);

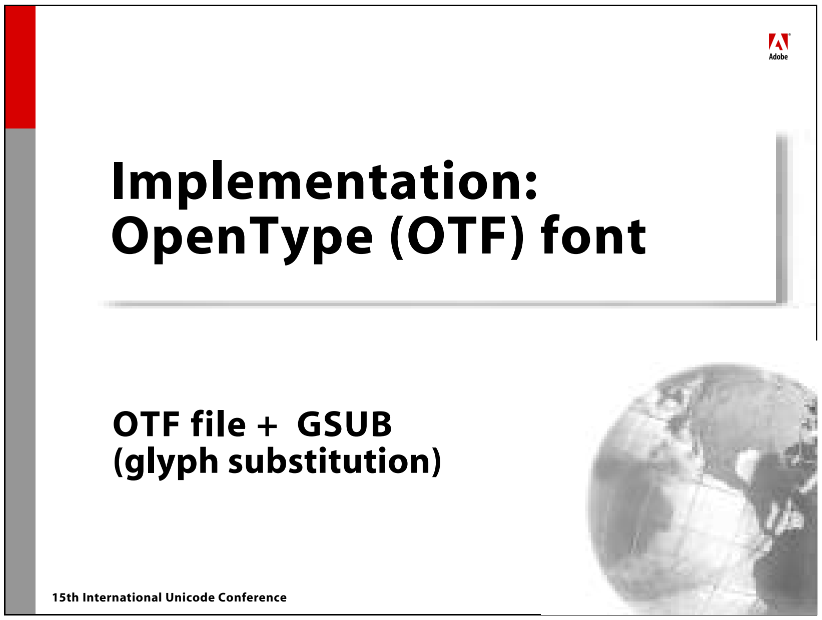
– 'zhTW-UCS2-H' (Traditional Chinese);

– 'jaJP-UCS2-H' (Japanese); and

– 'koKR-UCS2-H' (Korean).

[The names of the CMap files indicate language and country code, encoding, writing direction.]

[Note: A printout of the sample referred to on this slide will be handed out prior to the presentation.]



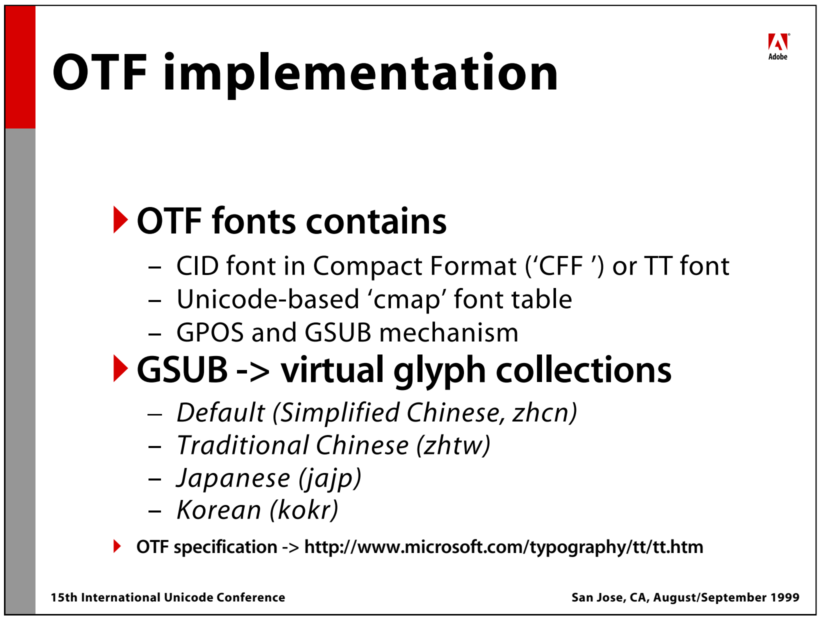
Implementation: OpenType (OTF) font

OTF file + GSUB (glyph substitution)

15th International Unicode Conference

A new development in the area of font technology is the OpenType font format. At a first glance, OpenType fonts do not offer the degree of openness or user-influence as fonts based on CID-keyed technology.

Big advantages of these Unicode-encoded fonts, however, are their cross-platform properties, a reduced file size (based on the Compact Font Format included in the font as the ‘CFF ’ table), and their advanced typographic features. These features allow for a huge variety of font-based modifications during the process of document creation. In the future, they will be supported by sophisticated publishing and layout applications.



OTF implementation

- OTF fonts contains
  - CID font in Compact Format (‘CFF ’) or TT font
  - Unicode-based ‘cmap’ font table
  - GPOS and GSUB mechanism
- GSUB -> virtual glyph collections
  - *Default (Simplified Chinese, zhcn)*
  - *Traditional Chinese (zhtw)*
  - *Japanese (jajp)*
  - *Korean (kokr)*
- OTF specification -> http://www.microsoft.com/typography/tt/tt.htm

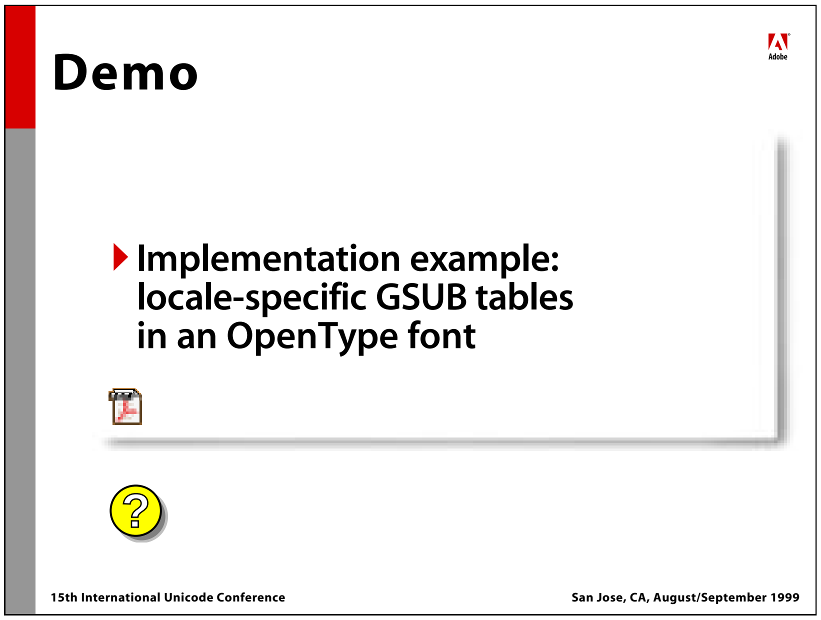
15th International Unicode Conference San Jose, CA, August/September 1999

In order to create an Pan-CJK Unicode font based on OTF technology, the task of addressing different glyphs per Unicode code point must be solved using the “glyph substitution” or “GSUB” mechanism OTF provides. This mechanism allows for defining “features” which – when invoked – use it to replace one or more glyphs by others.

While the glyphs within the font file are stored in much the same way as in a CID-keyed font (in fact, an OpenType font is a CID-keyed font in compacted form with added features and a Unicode-based ‘cmap’ table), every information about locale-specific glyph addressing can be found within the very same file.

For each locale in addition to the default, specific features are created that invoke the OTF-specific GSUB mechanism. The feature ‘jajp’[the name represents a combination of language and country code], for example, invokes the substitution of all default “non-Japanese” CJK glyphs with glyphs that are considered to be culturally adequate and typographically correct for Japanese writing.

The same happens when selecting the features ‘zhtw’ or ‘kokr.’ Invoking the ‘zhcn’-feature prompts switching back to the default glyphs: no substitution is taking place in this case.



Demo

- Implementation example: locale-specific GSUB tables in an OpenType font

15th International Unicode Conference San Jose, CA, August/September 1999

This example shows the different glyph substitutions that are taking place when locale-specific features in the OpenType font are invoked.

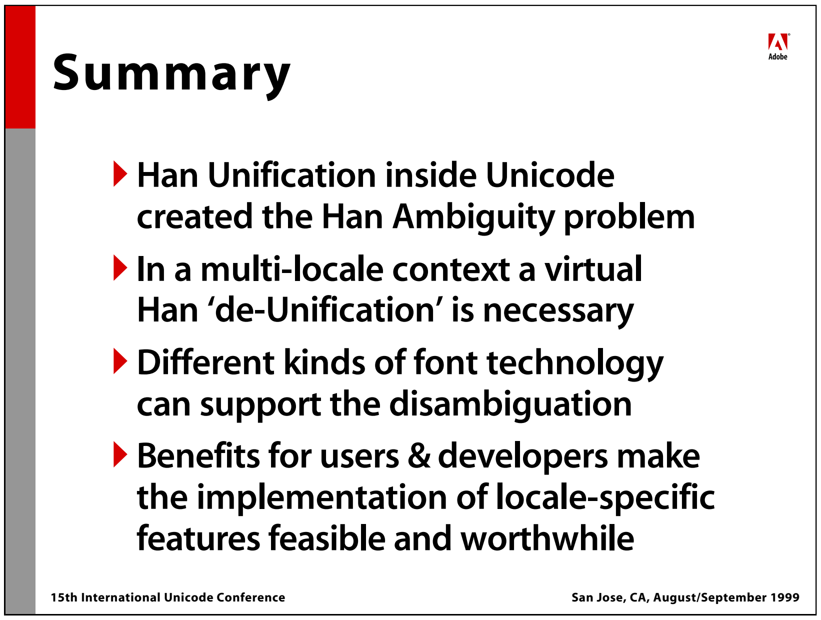
Where necessary, 'zhtw' substitutes the default glyphs with those different in the Traditional Chinese locale. The same is done by the features 'jajp' and 'kokr' for the Japanese and the Korean locale.

In this implementation, 'zhcn' is an "empty feature" that simply disables the others, thus effectively switching back to the font's default locale, Simplified Chinese.

Other approaches or features to implement cross-locale functionality are possible. For example, the single feature 'locl' can provide locale-specific glyph subsets that are invoked through different default "system languages" or user-influenced (through spell-checking, hyphenation) "application languages". The underlying glyph substitution mechanism, however, will not change.

The current OTF specification can be found at:
http://www.microsoft.com/typography/tt/tt.htm

[Note: A printout of the sample referred to on this slide will be handed out prior to the presentation.]



Summary

- Han Unification inside Unicode created the Han Ambiguity problem
- In a multi-locale context a virtual Han 'de-Unification' is necessary
- Different kinds of font technology can support the disambiguation
- Benefits for users & developers make the implementation of locale-specific features feasible and worthwhile

15th International Unicode Conference San Jose, CA, August/September 1999

In order to make all things presented here work in real-world situations, mechanisms like the ones described must be available and supported by applications. The first condition has already been met through advances in font technology. Hopefully, future applications will support fonts with cross-locale functionality for a long time to come.

When it comes to handling of CJK scripts in general, the approaches described here might prove to be especially satisfying: gradually, it becomes easier to handle the characters of different CJK locales according to locale-specific design rules and requirements. This is the level of functionality a lot of people have desired for a long time.

Adobe

Q & A

15th International Unicode Conference

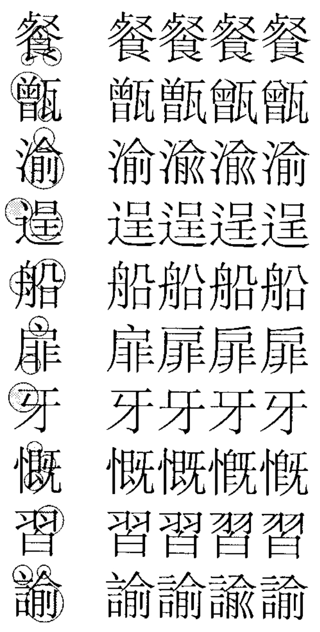
Dirk Meyer

dmeyer@adobe.com

Adobe Systems Inc.
CJKV Type Development
345 Park Avenue, M/S W8
San Jose, CA 95125, USA

Adobe

1 page. Back to main text? Click here !



餐	餐餐餐餐	*can*[1] – to eat, food, meal [U+9910]
甑	甑甑甑甑	*zeng*[4] – rice steamer [U+7511]
渝	渝渝渝渝	*yu*[2] – change (of attitude or feeling) [U+6E1D]
逞	逞逞逞逞	*cheng*[3] – to show off, flaunt, succeed (in a scheme) [U+901E]
船	船船船船	*chuan*[2] – boat, ship [U+823A]
扉	扉扉扉扉	*fei*[1] – door leaf [U+6249]
牙	牙牙牙牙	*ya*[2] – tooth, ivory [U+7259]
慨	慨慨慨慨	*kai*[3] – generous, deeply touched [U+6168]
習	習習習習	*xi*[2] – to practise, exercise, habit, custom [U+7FD2]
諭	諭諭諭諭	*yu*[4] – (old) instruct, tell [U+8AED]

This table provides examples that illustrate how subtle glyph differences (areas that show modifications are indicated by shaded circles) across locales can be. Note, how – according to the rules of Han Unification – one Unicode codepoint (U+) is used to represent four valid glyphs from four locales (G – T – J – K). "Han de-Unification" is absolutely necessary when correct glyphs for more than one locale are needed in one font.

1 page. Back to main text? Click here !

PanCJK U4exx (PanCJKU4exx)

一 c00G	丁 c01G	丂 c02G	丂 c02J	七 c03G	丄 c04G	丅 c05G	丆 c06G	万 c07G	丈 c08G	丈 c08T
三 c09G	上 c0AG	下 c0BG	丌 c0CG	不 c0DG	与 c0EG	与 c0EJ	与 c0ET	丏 c0FG	丏 c0FT	丐 c10G
丐 c10T	丑 c11G	丑 c11T	丒 c12G	专 c13G	且 c14G	丕 c15G	世 c16G	世 c16T	丗 c17G	丘 c18G
丙 c19G	业 c1AG	丛 c1BG	东 c1CG	丝 c1DG	丞 c1EG	丟 c1FG	丟 c1FT	丠 c20G	両 c21G	両 c21T
丢 c22G	丢 c22T	丣 c23G	丣 c23J	两 c24G	严 c25G	並 c26G	丧 c27G	丨 c28G	丩 c29G	个 c2AG
丫 c2BG	丬 c2CG	中 c2DG	丮 c2EG	丯 c2FG	丰 c30G	丰 c30T	丱 c31G	丱 c31T	串 c32G	丳 c33G
临 c34G	丵 c35G	丶 c36G	丷 c37G	丸 c38G	丸 c38T	丹 c39G	为 c3AG	主 c3BG	丼 c3CG	丽 c3DG
丽 c3DT	举 c3EG	丿 c3FG	乀 c40G	乁 c41G	乂 c42G	乂 c42J	乃 c43G	乃 c43T	乄 c44G	久 c45G
乆 c46G	乆 c46T	乇 c47G	么 c48G	么 c48T	义 c49G	乊 c4AG	之 c4BG	之 c4BT	乌 c4CG	乍 c4DG
乎 c4EG	乏 c4FG	乏 c4FT	乐 c50G	乑 c51G	乑 c51J	乒 c52G	乓 c53G	乔 c54G	乕 c55G	乖 c56G
乗 c57G	乘 c58G	乙 c59G	乚 c5AG	乛 c5BG	乜 c5CG	九 c5DG	乞 c5EG	也 c5FG	习 c60G	乡 c61G

This table illustrates the internal structure of a Song-style pan-CJK font using CID-keyed font technology. The glyph default for this font is the one of the "simplified Chinese" locale (G). Identical glyphs are not repeated, even if they are used in more than one locale, e.g. U+4E00, U+4E0D. Similarly, glyphs that differ across locales appear as necessary and in sequence, e.g. U+4E02, U+4E08, U+4E0E. The letters G, T, J, and K indicate which locale prefers the listed variant.

5 pages. Back to main text? Click here !

Adobe	一	丁	丂	丂	七	丄	丅	丆	万
0	1	2	3	4	5	6	7	8	9
丈	丈	三	上	下	丌	不	与	与	与
10	11	12	13	14	15	16	17	18	19
丏	丏	丏	丏	丑	丑	丒	专	且	丕
20	21	22	23	24	25	26	27	28	29
世	世	丗	丘	丙	业	丛	东	丝	丞
30	31	32	33	34	35	36	37	38	39
丟	丟	丠	両	両	丢	丢	丣	丣	两
40	41	42	43	44	45	46	47	48	49
严	並	丧	丨	丩	个	丫	丬	中	丮
50	51	52	53	54	55	56	57	58	59
丯	丰	丰	丱	丱	串	丳	临	丵	丶
60	61	62	63	64	65	66	67	68	69
丷	丸	丸	丹	为	主	丼	丽	丽	举
70	71	72	73	74	75	76	77	78	79
丿	乀	乁	乂	乂	乃	乃	乄	久	乆
80	81	82	83	84	85	86	87	88	89
乆	乇	么	么	义	乊	之	之	乌	乍
90	91	92	93	94	95	96	97	98	99
乎	乏	乏	乐	乑	乑	乒	乓	乔	乕
100	101	102	103	104	105	106	107	108	109
乖	乗	乘	乙	乚	乛	乜	九	乞	也
110	111	112	113	114	115	116	117	118	119

慧	慨	慨	慨	慨	慩	慩	慪	慪	慫
8040	41	8042	8043	8044	8045	8046	8047	8048	8049
慬	愁	慮	慯	慰	慱	慲	慲	慳	慳
8050	8051	8052	8053	8054	8055	8056	8057	8058	8059
慴	慴	慵	慵	慶	慶	慷	慷	慸	慹
8060	8061	8062	8063	8064	8065	8066	8067	8068	8069
慺	慺	慻	慻	慼	慽	慾	憑	憀	憀
8070	8071	8072	8073	8074	8075	8076	8077	8078	8079
憁	憁	憁	憂	憃	憃	德	德	憅	憆
8080	8081	8082	8083	8084	8085	8086	8087	8088	8089
憇	憈	憈	憉	憊	憊	憋	憋	慇	慇
8090	8091	8092	8093	8094	8095	8096	8097	8098	8099
憍	憎	憎	憎	憏	憐	憐	憐	憑	憒
8100	8101	8102	8103	8104	8105	8106	8107	8108	8109
憓	憔	憔	憕	憖	憖	憘	憙	憚	憛
8110	8111	8112	8113	8114	8115	8116	8117	8118	8119
憜	憝	憝	憞	憞	憟	憠	憠	憡	憢
8120	8121	8122	8123	8124	8125	8126	8127	8128	8129
憣	憤	憥	憦	憧	憧	憨	憨	憩	憪
8130	8131	8132	8133	8134	8135	8136	8137	8138	8139
憫	憫	憫	憬	憬	憭	憮	憯	憯	憯
8140	8141	8142	8143	8144	8145	8146	8147	8148	8149
憰	憰	憱	憱	憲	憲	憲	㦦	㦦	憴
8150	8151	8152	8153	8154	8155	8156	8157	8158	8159

戻 8400	扇 8401	扇 8402	扇 8403	扈 8404	扈 8405	扈 8406	扉 8407	扉 8408	扉 8409
扉 8410	扊 8411	扊 8412	手 8413	扌 8414	才 8415	才 8416	扎 8417	扏 8418	扐 8419
扑 8420	扒 8421	扒 8422	打 8423	扔 8424	扔 8425	払 8426	払 8427	扖 8428	扖 8429
扗 8430	托 8431	扙 8432	扙 8433	扚 8434	扚 8435	扛 8436	扜 8437	扝 8438	扝 8439
扞 8440	扟 8441	扟 8442	扠 8443	扠 8444	扡 8445	扢 8446	扣 8447	扤 8448	扥 8449
扦 8450	执 8451	扨 8452	扨 8453	扨 8454	扩 8455	扪 8456	扫 8457	扬 8458	扭 8459
扭 8460	扮 8461	扮 8462	扯 8463	扰 8464	扱 8465	扱 8466	扲 8467	扲 8468	扳 8469
扳 8470	扴 8471	扵 8472	扶 8473	扶 8474	扸 8475	批 8476	批 8477	批 8478	扺 8479
扺 8480	扻 8481	扼 8482	扽 8483	扽 8484	找 8485	承 8486	技 8487	抁 8488	抁 8489
抂 8490	抃 8491	抃 8492	抄 8493	抄 8494	抅 8495	抅 8496	抆 8497	抆 8498	抇 8499
抈 8500	抉 8501	把 8502	抋 8503	抌 8504	抍 8505	抎 8506	抎 8507	抏 8508	抐 8509
抐 8510	抑 8511	抑 8512	抒 8513	抓 8514	抔 8515	投 8516	投 8517	抖 8518	抗 8519

淔	淕	淕	淖	淘	淘	淘	淙	淙	淚
12960	12961	12962	12963	12964	12965	12966	12967	12968	12969
淚	淛	淜	淝	淞	淞	淟	淠	淡	淢
12970	12971	12972	12973	12974	12975	12976	12977	12978	12979
淣	淤	淤	淥	淥	淦	淧	淧	淨	淨
12980	12981	12982	12983	12984	12985	12986	12987	12988	12989
淩	淩	淪	淫	淫	淬	淬	淭	淭	淮
12990	12991	12992	12993	12994	12995	12996	12997	12998	12999
淮	淯	淯	淰	淰	深	深	淲	淲	淳
13000	13001	13002	13003	13004	13005	13006	13007	13008	13009
淳	淴	淵	淵	淶	混	混	淸	淹	淺
13010	13011	13012	13013	13014	13015	13016	13017	13018	13019
添	添	淼	淼	淽	淾	淿	渀	渁	渁
13020	13021	13022	13023	13024	13025	13026	13027	13028	13029
渂	渂	渃	渄	渄	渄	清	渆	渇	済
13030	13031	13032	13033	13034	13035	13036	13037	13038	13039
済	渉	渉	渊	渋	渌	渍	渎	渏	渐
13040	13041	13042	13043	13044	13045	13046	13047	13048	13049
渑	渒	渓	渔	渕	渖	渗	渗	渘	渙
13050	13051	13052	13053	13054	13055	13056	13057	13058	13059
渙	渚	渚	減	渜	渝	渝	渝	渝	渞
13060	13061	13062	13063	13064	065	13066	13067	13068	13069
渟	渟	渠	渠	渡	渡	渢	渢	渣	渣
13070	13071	13072	13073	13074	13075	13076	13077	13078	13079

爺 14760	爺 14761	爻 14762	爻 14763	爼 14764	爽 14765	爾 14766	爿 14767	爿 14768	牀 14769
牀 14770	牁 14771	牁 14772	牂 14773	牂 14774	牃 14775	牃 14776	牄 14777	牄 14778	牅 14779
牅 14780	牆 14781	牆 14782	片 14783	版 14784	版 14785	牉 14786	牉 14787	牊 14788	牋 14789
牌 14790	牌 14791	牍 14792	牎 14793	牏 14794	牏 14795	牏 14796	牐 14797	牑 14798	牑 14799
牒 14800	牒 14801	牓 14802	牓 14803	牔 14804	牕 14805	牕 14806	牕 14807	牖 14808	牖 14809
牗 14810	牗 14811	牘 14812	牘 14813	牙 ?814	牙 14815	牙 14816	牙 14817	牚 14818	牚 14819
牚 14820	牚 14821	牛 14822	牜 14823	牝 14824	牞 14825	牟 14826	牟 14827	牠 14828	牡 14829
牢 14830	牢 14831	牣 14832	牣 14833	牤 14834	牤 14835	牥 14836	牥 14837	牦 14838	牧 14839
牨 14840	牨 14841	物 14842	牪 14843	牫 14844	𤙁 14845	𤙁 14846	牭 14847	牭 14848	牮 14849
牯 14850	牰 14851	牱 14852	牲 14853	牳 14854	牳 14855	牴 14856	牴 14857	牵 14858	牶 14859
牶 14860	牷 14861	牷 14862	牸 14863	牸 14864	特 14865	牺 14866	牺 14867	牻 14868	牻 14869
牼 14870	牼 14871	牽 14872	牽 14873	牾 14874	牿 14875	牿 14876	犀 14877	犀 14878	犁 14879

8 pages. Back to main text? Click here !

Row 0x4E: STSongCJK-Light-zhCN-UCS2-H

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	一	丁	丂	七	丄	丅	丆	万	丈	三	上	下	丌	不	与	丏
1	丐	丑	丒	专	且	丕	世	丗	丘	丙	业	丛	东	丝	丞	丟
2	丠	両	丢	丣	两	严	並	丧	丨	丩	个	丫	丬	中	丮	丯
3	丰	丱	串	丳	临	丵	丶	丷	丸	丹	为	主	丼	丽	举	丿
4	乀	乁	乂	乃	乄	久	乆	乇	么	义	乊	之	乌	乍	乎	乏
5	乐	乑	乒	乓	乔	乕	乖	乗	乘	乙	乚	乛	乜	九	乞	也
6	习	乡	乢	乣	乤	乥	书	乧	乨	乩	乪	乫	乬	乭	乮	乯
7	买	乱	乲	乳	乴	乵	乶	乷	乸	乹	乺	乻	乼	乽	乾	乿
8	亀	亁	亂	亃	亄	亅	了	亇	予	争	亊	事	二	亍	于	亏
9	亐	云	互	亓	五	井	亖	亗	亘	亙	亚	些	亜	亝	亞	亟
A	亠	亡	亢	亣	交	亥	亦	产	亨	亩	亪	享	京	亭	亮	亯
B	亰	亱	亲	亳	亴	亵	亶	亷	亸	亹	人	亻	亼	亽	亾	亿
C	什	仁	仂	仃	仄	仅	仆	仇	仈	仉	今	介	仌	仍	从	仏
D	仐	仑	仒	仓	仔	仕	他	仗	付	仙	仚	仛	仜	仝	仞	仟
E	仠	仡	仢	代	令	以	仦	仧	仨	仩	仪	仫	们	仭	仮	仯
F	仰	仱	仲	仳	仴	仵	件	价	仸	仹	仺	任	仼	份	仾	仿

Row 0x9F: STSongCJK-Light-zhCN-UCS2-H

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	鼀	鼁	鼂	鼃	鼄	鼅	鼆	鼇	鼈	鼉	鼊	鼋	鼌	鼍	鼎	鼏
1	鼐	鼑	鼒	鼓	鼔	鼕	鼖	鼗	鼘	鼙	鼚	鼛	鼜	鼝	鼞	鼟
2	鼠	鼡	鼢	鼣	鼤	鼥	鼦	鼧	鼨	鼩	鼪	鼫	鼬	鼭	鼮	鼯
3	鼰	鼱	鼲	鼳	鼴	鼵	鼶	鼷	鼸	鼹	鼺	鼻	鼼	鼽	鼾	鼿
4	齀	齁	齂	齃	齄	齅	齆	齇	齈	齉	齊	齋	齌	齍	齎	齏
5	齐	齑	齒	齓	齔	齕	齖	齗	齘	齙	齚	齛	齜	齝	齞	齟
6	齠	齡	齢	齣	齤	齥	齦	齧	齨	齩	齪	齫	齬	齭	齮	齯
7	齰	齱	齲	齳	齴	齵	齶	齷	齸	齹	齺	齻	齼	齽	齾	齿
8	龀	龁	龂	龃	龄	龅	龆	龇	龈	龉	龊	龋	龌	龍	龎	龏
9	龐	龑	龒	龓	龔	龕	龖	龗	龘	龙	龚	龛	龜	龝	龞	龟
A	龠	龡	龢	龣	龤	龥	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
B	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
C	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
D	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
E	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
F	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe

Row 0x4E: STSongCJK-Light-zhTW-UCS2-H

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	一	丁	丂	七	丄	丅	丆	万	丈	三	上	下	丌	不	与	丏
1	丐	丑	丒	专	且	丕	世	丗	丘	丙	业	丛	东	丝	丞	丟
2	丠	両	丢	丣	两	严	並	丧	丨	丩	个	丫	丬	中	丮	丯
3	丰	丱	串	丳	临	丵	丶	丷	丸	丹	为	主	丼	丽	举	丿
4	乀	乁	乂	乃	乄	久	乆	乇	么	义	乊	之	乌	乍	乎	乏
5	乐	乑	乒	乓	乔	乕	乖	乗	乘	乙	乚	乛	乜	九	乞	也
6	习	乡	乢	乣	乤	乥	书	乧	乨	乩	乪	乫	乬	乭	乮	乯
7	买	乱	乲	乳	乴	乵	乶	乷	乸	乹	乺	乻	乼	乽	乾	乿
8	亀	亁	亂	亃	亄	亅	了	亇	予	争	亊	事	二	亍	于	亏
9	亐	云	互	亓	五	井	亖	亗	亘	亙	亚	些	亜	亝	亞	亟
A	亠	亡	亢	亣	交	亥	亦	产	亨	亩	亪	享	京	亭	亮	亯
B	亰	亱	亲	亳	亴	亵	亶	亷	亸	亹	人	亻	亼	亽	亾	亿
C	什	仁	仂	仃	仄	仅	仆	仇	仈	仉	今	介	仌	仍	从	仏
D	仐	仑	仒	仓	仔	仕	他	仗	付	仙	仚	仛	仜	仝	仞	仟
E	仠	仡	仢	代	令	以	仦	仧	仨	仩	仪	仫	们	仭	仮	仯
F	仰	仱	仲	仳	仴	仵	件	价	仸	仹	仺	任	仼	份	仾	仿

Row 0x9F: STSongCJK-Light-zhTW-UCS2-H

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	黿	鼁	鼂	鼃	鼄	鼅	鼆	鼇	鼈	鼉	鼊	鼋	鼌	鼍	鼎	鼏
1	鼐	鼑	鼒	鼓	鼔	鼕	鼖	鼗	鼘	鼙	鼚	鼛	鼜	鼝	鼞	鼟
2	鼠	鼡	鼢	鼣	鼤	鼥	鼦	鼧	鼨	鼩	鼪	鼫	鼬	鼭	鼮	鼯
3	鼰	鼱	鼲	鼳	鼴	鼵	鼶	鼷	鼸	鼹	鼺	鼻	鼼	鼽	鼾	鼿
4	齀	齁	齂	齃	齄	齅	齆	齇	齈	齉	齊	齋	齌	齍	齎	齏
5	齐	齑	齒	齓	齔	齕	齖	齗	齘	齙	齚	齛	齜	齝	齞	齟
6	齠	齡	齢	齣	齤	齥	齦	齧	齨	齩	齪	齫	齬	齭	齮	齯
7	齰	齱	齲	齳	齴	齵	齶	齷	齸	齹	齺	齻	齼	齽	齾	齿
8	龀	龁	龂	龃	龄	龅	龆	龇	龈	龉	龊	龋	龌	龍	龎	龏
9	龐	龑	龒	龓	龔	龕	龖	龗	龘	龙	龚	龛	龜	龝	龞	龟
A	龠	龡	龢	龣	龤	龥	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
B	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
C	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
D	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
E	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
F	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe

Row 0x4E: STSongCJK-Light-jaJP-UCS2-H

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	一	丁	丂	七	丄	丅	丆	万	丈	三	上	下	丌	不	与	丏
1	丐	丑	丒	专	且	丕	世	丗	丘	丙	业	丛	东	丝	丞	丟
2	丠	両	丢	丣	两	严	並	丧	丨	丩	个	丫	丬	中	丮	丯
3	丰	丱	串	丳	临	丵	丶	丷	丸	丹	为	主	丼	丽	举	丿
4	乀	乁	乂	乃	乄	久	乆	乇	么	义	乊	之	乌	乍	乎	乏
5	乐	乑	乒	乓	乔	乕	乖	乗	乘	乙	乚	乛	乜	九	乞	也
6	习	乡	乢	乣	乤	乥	书	乧	乨	乩	乪	乫	乬	乭	乮	乯
7	买	乱	乲	乳	乴	乵	乶	乷	乸	乹	乺	乻	乼	乽	乾	乿
8	亀	亁	亂	亃	亄	亅	了	亇	予	争	亊	事	二	亍	于	亏
9	亐	云	互	亓	五	井	亖	亗	亘	亙	亚	些	亜	亝	亞	亟
A	亠	亡	亢	亣	交	亥	亦	产	亨	亩	亪	享	京	亭	亮	亯
B	亰	亱	亲	亳	亴	亵	亶	亷	亸	亹	人	亻	亼	亽	亾	亿
C	什	仁	仂	仃	仄	仅	仆	仇	仈	仉	今	介	仌	仍	从	仏
D	仐	仑	仒	仓	仔	仕	他	仗	付	仙	仚	仛	仜	仝	仞	仟
E	仠	仡	仢	代	令	以	仦	仧	仨	仩	仪	仫	们	仭	仮	仯
F	仰	仱	仲	仳	仴	仵	件	价	仸	仹	仺	任	仼	份	仾	仿

Row 0x9F: STSongCJK-Light-jaJP-UCS2-H

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	黿	鼁	鼂	鼃	鼄	鼅	鼆	鼇	鼈	鼉	鼊	鼋	鼌	鼍	鼎	鼏
1	鼐	鼑	鼒	鼓	鼔	鼕	鼖	鼗	鼘	鼙	鼚	鼛	鼜	鼝	鼞	鼟
2	鼠	鼡	鼢	鼣	鼤	鼥	鼦	鼧	鼨	鼩	鼪	鼫	鼬	鼭	鼮	鼯
3	鼰	鼱	鼲	鼳	鼴	鼵	鼶	鼷	鼸	鼹	鼺	鼻	鼼	鼽	鼾	鼿
4	齀	齁	齂	齃	齄	齅	齆	齇	齈	齉	齊	齋	齌	齍	齎	齏
5	齐	齑	齒	齓	齔	齕	齖	齗	齘	齙	齚	齛	齜	齝	齞	齟
6	齠	齡	齢	齣	齤	齥	齦	齧	齨	齩	齪	齫	齬	齭	齮	齯
7	齰	齱	齲	齳	齴	齵	齶	齷	齸	齹	齺	齻	齼	齽	齾	齿
8	龀	龁	龂	龃	龄	龅	龆	龇	龈	龉	龊	龋	龌	龍	龎	龏
9	龐	龑	龒	龓	龔	龕	龖	龗	龘	龙	龚	龛	龜	龝	龞	龟
A	龠	龡	龢	龣	龤	龥	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
B	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
C	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
D	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
E	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
F	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe

Row 0x4E: STSongCJK-Light-koKR-UCS2-H

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	一	丁	丂	七	丄	丅	丆	万	丈	三	上	下	丌	不	与	丏
1	丐	丑	丒	专	且	丕	世	丗	丘	丙	业	丛	东	丝	丞	丟
2	丠	両	丢	丣	两	严	並	丧	丨	丩	个	丫	丬	中	丮	丯
3	丰	丱	串	丳	临	丵	丶	丷	丸	丹	为	主	丼	丽	举	丿
4	乀	乁	乂	乃	乄	久	乆	乇	么	义	乊	之	乌	乍	乎	乏
5	乐	乑	乒	乓	乔	乕	乖	乗	乘	乙	乚	乛	乜	九	乞	也
6	习	乡	乢	乣	乤	乥	书	乧	乨	乩	乪	乫	乬	乭	乮	乯
7	买	乱	乲	乳	乴	乵	乶	乷	乸	乹	乺	乻	乼	乽	乾	乿
8	亀	亁	亂	亃	亄	亅	了	亇	予	争	亊	事	二	亍	于	亏
9	亐	云	互	亓	五	井	亖	亗	亘	亙	亚	些	亜	亝	亞	亟
A	亠	亡	亢	亣	交	亥	亦	产	亨	亩	亪	享	京	亭	亮	亯
B	亰	亱	亲	亳	亴	亵	亶	亷	亸	亹	人	亻	亼	亽	亾	亿
C	什	仁	仂	仃	仄	仅	仆	仇	仈	仉	今	介	仌	仍	从	仏
D	仐	仑	仒	仓	仔	仕	他	仗	付	仙	仚	仛	仜	仝	仞	仟
E	仠	仡	仢	代	令	以	仦	仧	仨	仩	仪	仫	们	仭	仮	仯
F	仰	仱	仲	仳	仴	仵	件	价	仸	仹	仺	任	仼	份	仾	仿

Row 0x9F: STSongCJK-Light-koKR-UCS2-H

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	鼀	鼁	鼂	鼃	鼄	鼅	鼆	鼇	鼈	鼉	鼊	鼋	鼌	鼍	鼎	鼏
1	鼐	鼑	鼒	鼓	鼔	鼕	鼖	鼗	鼘	鼙	鼚	鼛	鼜	鼝	鼞	鼟
2	鼠	鼡	鼢	鼣	鼤	鼥	鼦	鼧	鼨	鼩	鼪	鼫	鼬	鼭	鼮	鼯
3	鼰	鼱	鼲	鼳	鼴	鼵	鼶	鼷	鼸	鼹	鼺	鼻	鼼	鼽	鼾	鼿
4	齀	齁	齂	齃	齄	齅	齆	齇	齈	齉	齊	齋	齌	齍	齎	齏
5	齐	齑	齒	齓	齔	齕	齖	齗	齘	齙	齚	齛	齜	齝	齞	齟
6	齠	齡	齢	齣	齤	齥	齦	齧	齨	齩	齪	齫	齬	齭	齮	齯
7	齰	齱	齲	齳	齴	齵	齶	齷	齸	齹	齺	齻	齼	齽	齾	齿
8	龀	龁	龂	龃	龄	龅	龆	龇	龈	龉	龊	龋	龌	龍	龎	龏
9	龐	龑	龒	龓	龔	龕	龖	龗	龘	龙	龚	龛	龜	龝	龞	龟
A	龠	龡	龢	龣	龤	龥	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
B	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
C	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
D	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
E	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe
F	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe	Adobe

6 pages. Back to main text? Click here !

丈	→	丈	与	→	与	丏	→	丏
\00010@10		\00011@11	\00017@17		\00018@18	\00020@20		\00021@21
丏	→	丏	丑	→	丑	世	→	世
\00022@22		\00023@23	\00024@24		\00025@25	\00030@30		\00031@31
丟	→	丟	両	→	両	丢	→	丢
\00040@40		\00041@41	\00043@43		\00044@44	\00045@45		\00046@46
丰	→	丰	丱	→	丱	丸	→	丸
\00061@61		\00062@62	\00063@63		\00064@64	\00071@71		\00072@72
丽	→	丽	乃	→	乃	乆	→	乆
\00077@77		\00078@78	\00085@85		\00086@86	\00089@89		\00090@90
么	→	么	之	→	之	乏	→	乏
\00092@92		\00093@93	\00096@96		\00097@97	\00101@101		\00102@102
乢	→	乢	乣	→	乣	乱	→	乱
\00122@122		\00123@123	\00124@124		\00125@125	\00130@130		\00131@131
乳	→	乳	乿	→	乿	乾	→	乾
\00143@143		\00144@144	\00159@159		\00160@160	\00162@162		\00163@163
亂	→	亂	亃	→	亃	亄	→	亄
\00164@164		\00165@165	\00166@166		\00167@167	\00168@168		\00169@169
亇	→	亇	亐	→	亐	云	→	云
\00172@172		\00173@173	\00181@181		\00182@182	\00185@185		\00186@186
互	→	互	亗	→	亗	亝	→	亝
\00187@187		\00188@188	\00193@193		\00194@194	\00200@200		\00201@201

齬 \33810@33810	→	齬 \33811@33811	齮 \33814@33814	→	齮 \33815@33815	齯 \33817@33817	→	齯 \33818@33818
齰 \33819@33819	→	齰 \33820@33820	齱 \33821@33821	→	齱 \33822@33822	齲 \33823@33823	→	齲 \33824@33824
齴 \33828@33828	→	齴 \33829@33829	齵 \33830@33830	→	齵 \33831@33831	齶 \33833@33833	→	齶 \33834@33834
齷 \33835@33835	→	齷 \33836@33836	齸 \33837@33837	→	齸 \33838@33838	齹 \33839@33839	→	齹 \33840@33840
齺 \33841@33841	→	齺 \33842@33842	齻 \33843@33843	→	齻 \33844@33844	齼 \33845@33845	→	齼 \33846@33846
齾 \33849@33849	→	齾 \33850@33850	龍 \33865@33865	→	龍 \33866@33866	龐 \33867@33867	→	龐 \33868@33868
龐 \33871@33871	→	龐 \33872@33872	龔 \33873@33873	→	龔 \33874@33874	龒 \33875@33875	→	龒 \33876@33876
櫳 \33877@33877	→	櫳 \33878@33878	龔 \33879@33879	→	龔 \33880@33880	龕 \33881@33881	→	龕 \33882@33882
龗 \33885@33885	→	龗 \33886@33886	龘 \33887@33887	→	龘 \33888@33888	龜 \33892@33892	→	龜 \33893@33893
穐 \33894@33894	→	穐 \33895@33895	龣 \33903@33903	→	龣 \33904@33904	龤 \33905@33905	→	龤 \33906@33906

丂	→	丂	与	→	与	丏	→	丏
\00003@3		\00004@4	\00017@17		\00019@19	\00022@22		\00023@23
丑	→	丑	世	→	世	丟	→	丟
\00024@24		\00025@25	\00030@30		\00031@31	\00040@40		\00041@41
两	→	两	丣	→	丣	丱	→	丱
\00043@43		\00044@44	\00047@47		\00048@48	\00063@63		\00064@64
乂	→	乂	乃	→	乃	之	→	之
\00083@83		\00084@84	\00085@85		\00086@86	\00096@96		\00097@97
乏	→	乏	乑	→	乑	乢	→	乢
\00101@101		\00102@102	\00104@104		\00105@105	\00122@122		\00123@123
乣	→	乣	乨	→	乨	乳	→	乳
\00124@124		\00125@125	\00130@130		\00131@131	\00146@146		\00147@147
乿	→	乿	亂	→	亂	云	→	云
\00159@159		\00160@160	\00164@164		\00165@165	\00185@185		\00186@186
互	→	互	亗	→	亗	亝	→	亝
\00187@187		\00188@188	\00193@193		\00194@194	\00200@200		\00201@201
亞	→	亞	亟	→	亟	亠	→	亠
\00202@202		\00203@203	\00204@204		\00205@205	\00206@206		\00207@207
亡	→	亡	亢	→	亢	交	→	交
\00208@208		\00210@210	\00211@211		\00212@212	\00215@215		\00217@217
亥	→	亥	亦	→	亦	亨	→	亨
\00218@218		\00219@219	\00220@220		\00221@221	\00223@223		\00224@224

齋	→	齋	齎	→	齎	齏	→	齏
\33743@33743		\33744@33744	\33749@33749		\33750@33750	\33751@33751		\33752@33752
齒	→	齒	齓	→	齓	齔	→	齔
\33756@33756		\33757@33757	\33758@33758		\33759@33759	\33760@33760		\33761@33761
齕	→	齕	齖	→	齖	齗	→	齗
\33762@33762		\33763@33763	\33764@33764		\33766@33766	\33767@33767		\33768@33768
齘	→	齘	齚	→	齚	齝	→	齝
\33769@33769		\33770@33770	\33773@33773		\33774@33774	\33779@33779		\33780@33780
齞	→	齞	齟	→	齟	齠	→	齠
\33781@33781		\33782@33782	\33783@33783		\33784@33784	\33785@33785		\33786@33786
齡	→	齡	齡	→	齡	齣	→	齣
\33787@33787		\33788@33788	\33789@33789		\33790@33790	\33791@33791		\33792@33792
齦	→	齦	齧	→	齧	齨	→	齨
\33797@33797		\33798@33798	\33799@33799		\33800@33800	\33801@33801		\33802@33802
齩	→	齩	齪	→	齪	齬	→	齬
\33803@33803		\33804@33804	\33806@33806		\33807@33807	\33810@33810		\33811@33811
齭	→	齭	齮	→	齮	齯	→	齯
\33812@33812		\33813@33813	\33814@33814		\33816@33816	\33817@33817		\33818@33818
齰	→	齰	齱	→	齱	齲	→	齲
\33819@33819		\33820@33820	\33821@33821		\33822@33822	\33823@33823		\33825@33825
齳	→	齳	齵	→	齵	齶	→	齶
\33826@33826		\33827@33827	\33830@33830		\33832@33832	\33833@33833		\33834@33834

丈	→	丈		丑	→	丑	
\00010@10		\00011@11		\00024@24		\00025@25	
世	→	世		丟	→	丟	
\00030@30		\00031@31		\00040@40		\00041@41	
丱	→	丱		乂	→	乂	
\00063@63		\00064@64		\00083@83		\00084@84	

丈	→	丈	丑	→	丑	世	→	世
\00010@10		\00011@11	\00024@24		\00025@25	\00030@30		\00031@31
丟	→	丟	丱	→	丱	乂	→	乂
\00040@40		\00041@41	\00063@63		\00064@64	\00083@83		\00084@84
乃	→	乃	之	→	之	乏	→	乏
\00085@85		\00086@86	\00096@96		\00097@97	\00101@101		\00102@102
乬	→	乬	乳	→	乳	乷	→	乷
\00135@135		\00136@136	\00143@143		\00144@144	\00149@149		\00150@150
乼	→	乼	亂	→	亂	亏	→	亏
\00155@155		\00156@156	\00164@164		\00165@165	\00181@181		\00182@182
亐	→	亐	云	→	云	互	→	互
\00183@183		\00184@184	\00185@185		\00186@186	\00187@187		\00188@188
亞	→	亞	亟	→	亟	亡	→	亡
\00202@202		\00203@203	\00204@204		\00205@205	\00208@208		\00210@210
亢	→	亢	交	→	交	亥	→	亥
\00211@211		\00212@212	\00215@215		\00217@217	\00218@218		\00219@219
亦	→	亦	亨	→	亨	享	→	享
\00220@220		\00221@221	\00223@223		\00224@224	\00227@227		\00228@228
京	→	京	亭	→	亭	亮	→	亮
\00229@229		\00230@230	\00231@231		\00232@232	\00233@233		\00234@234
亳	→	亳	亴	→	亴	亶	→	亶
\00243@243		\00244@244	\00245@245		\00246@246	\00248@248		\00249@249

鼇	→	鼇	鼈	→	鼈	鼉	→	鼉
\33607@33607		\33609@33609	\33610@33610		\33612@33612	\33614@33614		\33615@33615
鼎	→	鼎	鼐	→	鼐	鼠	→	鼠
\33619@33619		\33621@33621	\33624@33624		\33625@33625	\33654@33654		\33656@33656
鼢	→	鼢	鼫	→	鼫	鼬	→	鼬
\33658@33658		\33659@33659	\33679@33679		\33680@33680	\33682@33682		\33683@33683
鼯	→	鼯	鼴	→	鼴	鼸	→	鼸
\33688@33688		\33689@33689	\33701@33701		\33702@33702	\33711@33711		\33712@33712
鼹	→	鼹	鼻	→	鼻	齊	→	齊
\33713@33713		\33714@33714	\33718@33718		\33719@33719	\33741@33741		\33742@33742
齋	→	齋	齎	→	齎	齒	→	齒
\33743@33743		\33744@33744	\33749@33749		\33750@33750	\33756@33756		\33757@33757
齔	→	齔	齕	→	齕	齗	→	齗
\33760@33760		\33761@33761	\33762@33762		\33763@33763	\33767@33767		\33768@33768
齟	→	齟	齠	→	齠	齡	→	齡
\33783@33783		\33784@33784	\33785@33785		\33786@33786	\33787@33787		\33788@33788
齦	→	齦	齧	→	齧	齩	→	齩
\33797@33797		\33798@33798	\33799@33799		\33800@33800	\33803@33803		\33805@33805
齪	→	齪	齫	→	齫	齬	→	齬
\33806@33806		\33807@33807	\33808@33808		\33809@33809	\33810@33810		\33811@33811
齲	→	齲	齶	→	齶	齷	→	齷
\33823@33823		\33825@33825	\33833@33833		\33834@33834	\33835@33835		\33836@33836