Pan-CJK Font Development Techniques, Tips, Tricks & Pitfalls Dr. Ken Lunde | Sr Computer Scientist 2 | CJKV Type Development | Adobe Systems Incorporated @ken_lunde http://adobe.ly/1R3YbuW 嚴祖祖 -82-「「」、けけ ムッケッケシ 该该汤汤场 *其*R 亢 R 分 R 分 R 方 R 方 R 开 R 尹 R 斤 **聚了**雷 廣了廣了輦「賛了 督團星 「小学校」



Pan-CJK Font Development





• Due to the large number of necessary code points, the number of required glyphs is extraordinarily high

- Due to the large number of necessary code points, the number of required glyphs is extraordinarily high
 - Unicode Version 8.0 includes 80,388 CJK Unified Ideographs and over 120K characters



- Due to the large number of necessary code points, the number of required glyphs is extraordinarily high
 - Unicode Version 8.0 includes 80,388 CJK Unified Ideographs and over 120K characters
- As the number of glyphs increases, so does the number of Unicode mappings and OpenType features



- Due to the large number of necessary code points, the number of required glyphs is extraordinarily high
 - Unicode Version 8.0 includes 80,388 CJK Unified Ideographs and over 120K characters
- As the number of glyphs increases, so does the number of Unicode mappings and OpenType features
- A large number of characters have multiple language- or region-specific forms
 - Mainly CJK Unified Ideographs, but some punctuation and symbols are also affected

- Due to the large number of necessary code points, the number of required glyphs is extraordinarily high
 - Unicode Version 8.0 includes 80,388 CJK Unified Ideographs and over 120K characters
- As the number of glyphs increases, so does the number of Unicode mappings and OpenType features
- A large number of characters have multiple language- or region-specific forms
 - Mainly CJK Unified Ideographs, but some punctuation and symbols are also affected

- Due to the large number of necessary code points, the number of required glyphs is extraordinarily high
 - Unicode Version 8.0 includes 80,388 CJK Unified Ideographs and over 120K characters
- As the number of glyphs increases, so does the number of Unicode mappings and OpenType features
- A large number of characters have multiple language- or region-specific forms
 - Mainly CJK Unified Ideographs, but some punctuation and symbols are also affected

字 숙 字

- Due to the large number of necessary code points, the number of required glyphs is extraordinarily high
 - Unicode Version 8.0 includes 80,388 CJK Unified Ideographs and over 120K characters
- As the number of glyphs increases, so does the number of Unicode mappings and OpenType features
- A large number of characters have multiple language- or region-specific forms
 - Mainly CJK Unified Ideographs, but some punctuation and symbols are also affected

字与字骨

- Due to the large number of necessary code points, the number of required glyphs is extraordinarily high
 - Unicode Version 8.0 includes 80,388 CJK Unified Ideographs and over 120K characters
- As the number of glyphs increases, so does the number of Unicode mappings and OpenType features
- A large number of characters have multiple language- or region-specific forms
 - Mainly CJK Unified Ideographs, but some punctuation and symbols are also affected

- Due to the large number of necessary code points, the number of required glyphs is extraordinarily high
 - Unicode Version 8.0 includes 80,388 CJK Unified Ideographs and over 120K characters
- As the number of glyphs increases, so does the number of Unicode mappings and OpenType features
- A large number of characters have multiple language- or region-specific forms
 - Mainly CJK Unified Ideographs, but some punctuation and symbols are also affected



- Due to the large number of necessary code points, the number of required glyphs is extraordinarily high
 - Unicode Version 8.0 includes 80,388 CJK Unified Ideographs and over 120K characters
- As the number of glyphs increases, so does the number of Unicode mappings and OpenType features
- A large number of characters have multiple language- or region-specific forms
 - Mainly CJK Unified Ideographs, but some punctuation and symbols are also affected
- Lots of pieces + Lots of moving parts → **Great potential for errors**

- Due to the large number of necessary code points, the number of required glyphs is extraordinarily high
 - Unicode Version 8.0 includes 80,388 CJK Unified Ideographs and over 120K characters
- As the number of glyphs increases, so does the number of Unicode mappings and OpenType features
- A large number of characters have multiple language- or region-specific forms
 - Mainly CJK Unified Ideographs, but some punctuation and symbols are also affected
- Lots of pieces + Lots of moving parts → Great potential for errors
- Therefore \rightarrow automate and script as many of the processes as possible

In other words → Pan-CJK font development...



- In other words → Pan-CJK font development...
 - ...is complex

- In other words → Pan-CJK font development...
 - ...is complex
 - ...requires a lot of work

- In other words → Pan-CJK font development...
 - ...is complex
 - ...requires a lot of work
 - ... is prone to error

- In other words → Pan-CJK font development...
 - ...is complex
 - ...requires a lot of work
 - ...is prone to error
 - ...and has many hurdles

- In other words → Pan-CJK font development...
 - ...is complex
 - ...requires a lot of work
 - ...is prone to error
 - ...and has many hurdles
- But the end result is a consistent typeface design that supports multiple East Asian languages and regions

思源黑体 思源黑體 源ノ角ゴシック 본고딕 思源黑体 思源黑體 源ノ角ゴシック 본고딕

Pan-CJK Font Development **Techniques**



• Not all CJK Unified Ideographs need to be included!



- Not all CJK Unified Ideographs need to be included!
- CJK Unified Ideographs → Coverage is best determined by regional standard coverage

- Not all CJK Unified Ideographs need to be included!
- CJK Unified Ideographs → Coverage is best determined by regional standard coverage
 - China (PRC) → GB 18030 & TGH 2013 (通用规范汉字表)



- Not all CJK Unified Ideographs need to be included!
- CJK Unified Ideographs \rightarrow Coverage is best determined by regional standard coverage
 - China (PRC) → GB 18030 & TGH 2013 (通用规范汉字表)
 - Taiwan (ROC) \rightarrow Big Five & CNS 11643 (Planes 1 & 2 at a minimum)



(共 524 頁

- Not all CJK Unified Ideographs need to be included!
- CJK Unified Ideographs → Coverage is best determined by regional standard coverage
 - China (PRC) → GB 18030 & TGH 2013 (通用规范汉字表)
 - Taiwan (ROC) → Big Five & CNS 11643 (Planes 1 & 2 at a minimum)
 - Hong Kong SAR \rightarrow HKSCS (2008) \rightarrow HKCS (2015)



- Not all CJK Unified Ideographs need to be included!
- CJK Unified Ideographs → Coverage is best determined by regional standard coverage
 - China (PRC) → GB 18030 & TGH 2013 (通用规范汉字表)
 - Taiwan (ROC) → Big Five & CNS 11643 (Planes 1 & 2 at a minimum)
 - Hong Kong SAR \rightarrow HKSCS (2008) \rightarrow HKCS (2015)
 - Japan \rightarrow JIS X 0208, JIS X 0212 & JIS X 0213



Adobe

- Not all CJK Unified Ideographs need to be included!
- CJK Unified Ideographs → Coverage is best determined by regional standard coverage
 - China (PRC) → GB 18030 & TGH 2013 (通用规范汉字表)
 - Taiwan (ROC) → Big Five & CNS 11643 (Planes 1 & 2 at a minimum)
 - Hong Kong SAR \rightarrow HKSCS (2008) \rightarrow HKCS (2015)
 - Japan \rightarrow JIS X 0208, JIS X 0212 & JIS X 0213
 - Korea (ROK) → KS X 1001 & KS X 1002



Α

- Not all CJK Unified Ideographs need to be included!
- CJK Unified Ideographs → Coverage is best determined by regional standard coverage
 - China (PRC) → GB 18030 & TGH 2013 (通用规范汉字表)
 - Taiwan (ROC) → Big Five & CNS 11643 (Planes 1 & 2 at a minimum)
 - Hong Kong SAR \rightarrow HKSCS (2008) \rightarrow HKCS (2015)
 - Japan \rightarrow JIS X 0208, JIS X 0212 & JIS X 0213
 - Korea (ROK) → KS X 1001 & KS X 1002
- Hangul → All modern syllables at a minimum—consider support for combining jamo



- Not all CJK Unified Ideographs need to be included!
- CJK Unified Ideographs → Coverage is best determined by regional standard coverage
 - China (PRC) → GB 18030 & TGH 2013 (通用规范汉字表)
 - Taiwan (ROC) → Big Five & CNS 11643 (Planes 1 & 2 at a minimum)
 - Hong Kong SAR \rightarrow HKSCS (2008) \rightarrow HKCS (2015)
 - Japan \rightarrow JIS X 0208, JIS X 0212 & JIS X 0213
 - Korea (ROK) → KS X 1001 & KS X 1002
- Hangul \rightarrow All modern syllables at a minimum—consider support for combining jamo
- Latin \rightarrow ASCII plus characters that are useful for CJK transcription & transliteration purposes

- Not all CJK Unified Ideographs need to be included!
- CJK Unified Ideographs → Coverage is best determined by regional standard coverage
 - China (PRC) → GB 18030 & TGH 2013 (通用规范汉字表)
 - Taiwan (ROC) → Big Five & CNS 11643 (Planes 1 & 2 at a minimum)
 - Hong Kong SAR \rightarrow HKSCS (2008) \rightarrow HKCS (2015)
 - Japan \rightarrow JIS X 0208, JIS X 0212 & JIS X 0213
 - Korea (ROK) \rightarrow KS X 1001 & KS X 1002
- Hangul \rightarrow All modern syllables at a minimum—consider support for combining jamo
- Latin \rightarrow ASCII plus characters that are useful for CJK transcription & transliteration purposes
 - CJKV? Consider supporting Latin-based Vietnamese—Quốc ngữ

- Not all CJK Unified Ideographs need to be included!
- CJK Unified Ideographs → Coverage is best determined by regional standard coverage
 - China (PRC) → GB 18030 & TGH 2013 (通用规范汉字表)
 - Taiwan (ROC) → Big Five & CNS 11643 (Planes 1 & 2 at a minimum)
 - Hong Kong SAR \rightarrow HKSCS (2008) \rightarrow HKCS (2015)
 - Japan \rightarrow JIS X 0208, JIS X 0212 & JIS X 0213
 - Korea (ROK) → KS X 1001 & KS X 1002
- Hangul \rightarrow All modern syllables at a minimum—consider support for combining jamo
- Latin \rightarrow ASCII plus characters that are useful for CJK transcription & transliteration purposes
 - CJKV? Consider supporting Latin-based Vietnamese—*Quốc ngữ*
- Symbols → Based on supported regional standards—consider support for entire CJK-related blocks

• Don't let the tail wag the dog!

- Don't let the tail wag the dog!
- Consider the primary script of the font

- Don't let the tail wag the dog!
- Consider the primary script of the font
- "Pan-CJK" effectively means that CJK Unified Ideographs represent the primary script

- Don't let the tail wag the dog!
- Consider the primary script of the font
- "Pan-CJK" effectively means that CJK Unified Ideographs represent the primary script
- Other scripts are adjusted to harmonize with the primary script
 - Especially Latin
Technique #2 — Harmonizing Glyphs For Different Scripts

- Don't let the tail wag the dog!
- Consider the primary script of the font
- "Pan-CJK" effectively means that CJK Unified Ideographs represent the primary script
- Other scripts are adjusted to harmonize with the primary script
 - Especially Latin
- Harmonization is usually performed via scaling & baseline shifting

Technique #2 — Harmonizing Glyphs For Different Scripts

- Don't let the tail wag the dog!
- Consider the primary script of the font
- "Pan-CJK" effectively means that CJK Unified Ideographs represent the primary script
- Other scripts are adjusted to harmonize with the primary script
 - Especially Latin
- Harmonization is usually performed via scaling & baseline shifting

Source Han Sans / Noto Sans CJK Source Sans Pro

Unicode Version 8.0 ↔ Unicode Version 8.0

Unicode Version 8.0 ↔ Unicode Version 8.0 Unicode Version 8.0 ↔ Unicode Version 8.0

Technique #2 — Harmonizing Glyphs For Different Scripts

- Don't let the tail wag the dog!
- Consider the primary script of the font
- "Pan-CJK" effectively means that CJK Unified Ideographs represent the primary script
- Other scripts are adjusted to harmonize with the primary script
 - Especially Latin
- Harmonization is usually performed via scaling & baseline shifting

Source Han Sans HW / Noto Sans Mono CJK / 'hwid' Source Code Pro

Unicode Version 8.0 ↔ Unicode Version 8.0

Unicode Version 8.0 ↔ Unicode Version 8.0 Unicode Version 8.0 ↔ Unicode Version 8.0

• The purpose of the 'locl' GSUB feature is to *mimic* the 'cmap' table for non-default languages

• The purpose of the 'locl' GSUB feature is to *mimic* the 'cmap' table for non-default languages



- The purpose of the 'locl' GSUB feature is to *mimic* the 'cmap' table for non-default languages
- A Pan-CJK font requires a default language
 - Other supported—but non-default—languages require 'locl' GSUB feature lookups

- The purpose of the 'locl' GSUB feature is to *mimic* the 'cmap' table for non-default languages
- A Pan-CJK font requires a default language
 - Other supported—but non-default—languages require 'locl' GSUB feature lookups
- A Perl script was developed that uses **default language** & **target language** CMap resources as arguments
 - Output \rightarrow Substitutions that change **default language** 'cmap' table mappings into **target language**

- The purpose of the 'locl' GSUB feature is to *mimic* the 'cmap' table for non-default languages
- A Pan-CJK font requires a default language
 - Other supported—but non-default—languages require 'locl' GSUB feature lookups
- A Perl script was developed that uses **default language** & **target language** CMap resources as arguments
 - Output → Substitutions that change default language 'cmap' table mappings into target language
- NOTE: CMap resource changes have the potential to necessitate re-running this script for each language pair

Better hinting means improved on-screen rendering

- Better hinting means improved on-screen rendering
- Hinting parameters \rightarrow horizontal stems, vertical stems & alignment zones

- Better hinting means improved on-screen rendering
- Hinting parameters \rightarrow horizontal stems, vertical stems & alignment zones



- Better hinting means improved on-screen rendering
- Hinting parameters \rightarrow horizontal stems, vertical stems & alignment zones
- Establish glyph classes whose glyphs benefit from common hinting parameters
 - Ideographs, Hangul, Kana, Bopomofo, Latin, Digits, Symbols, and so on

- Better hinting means improved on-screen rendering
- Hinting parameters \rightarrow horizontal stems, vertical stems & alignment zones
- Establish glyph classes whose glyphs benefit from common hinting parameters
 - Ideographs, Hangul, Kana, Bopomofo, Latin, Digits, Symbols, and so on
- Each glyph class has its own FDArray element with its own hinting parameters

- Better hinting means improved on-screen rendering
- Hinting parameters \rightarrow horizontal stems, vertical stems & alignment zones
- Establish glyph classes whose glyphs benefit from common hinting parameters
 - Ideographs, Hangul, Kana, Bopomofo, Latin, Digits, Symbols, and so on
- Each glyph class has its own FDArray element with its own hinting parameters
- The implementation limit for FDArray elements is 256



- Better hinting means improved on-screen rendering
- Hinting parameters \rightarrow horizontal stems, vertical stems & alignment zones
- Establish glyph classes whose glyphs benefit from common hinting parameters
 - Ideographs, Hangul, Kana, Bopomofo, Latin, Digits, Symbols, and so on
- Each glyph class has its own FDArray element with its own hinting parameters
- The implementation limit for FDArray elements is 256
 - Source Han Sans / Noto Sans CJK uses 19 FDArray elements



Pan-CJK Font Development Tips & Tricks





- Unicode-based working glyph names can establish unambiguous and unique identity
 - uniXXXX (BMP) or uXXXXX/uXXXXXX (Planes 1 through 16)

- Unicode-based working glyph names can establish unambiguous and unique identity
 - uniXXXX (BMP) or uXXXXX/uXXXXXX (Planes 1 through 16)
- Specify sequences—kana, variation sequences, high-frequency archaic hangul syllables
 - Concatenations of Unicode-based working glyph names

- Unicode-based working glyph names can establish unambiguous and unique identity
 - uniXXXX (BMP) or uXXXXX/uXXXXXX (Planes 1 through 16)
- Specify sequences—kana, variation sequences, high-frequency archaic hangul syllables
 - Concatenations of Unicode-based working glyph names





- Unicode-based working glyph names can establish unambiguous and unique identity
 - uniXXXX (BMP) or uXXXXX/uXXXXXX (Planes 1 through 16)
- Specify sequences—kana, variation sequences, high-frequency archaic hangul syllables
 - Concatenations of Unicode-based working glyph names
- Use of additional identifiers in Unicode-based working glyph names to...

- Unicode-based working glyph names can establish unambiguous and unique identity
 - uniXXXX (BMP) or uXXXXX/uXXXXXX (Planes 1 through 16)
- Specify sequences—kana, variation sequences, high-frequency archaic hangul syllables
 - Concatenations of Unicode-based working glyph names
- Use of additional identifiers in Unicode-based working glyph names to...
 - ...specify languages—CN, HK, JP, KR, & TW
 - ...specify glyph width—FW, HW, PW, QW, & TW
 - ...specify vertical variants—V
 - ...specify particular GSUB features—JP90, ljmoXX, vjmoXX & tjmoXX

- Unicode-based working glyph names can establish unambiguous and unique identity
 - uniXXXX (BMP) or uXXXXX/uXXXXXX (Planes 1 through 16)
- Specify sequences—kana, variation sequences, high-frequency archaic hangul syllables
 - Concatenations of Unicode-based working glyph names
- Use of additional identifiers in Unicode-based working glyph names to...
 - ...specify languages—CN, HK, JP, KR, & TW
 - ...specify glyph width—FW, HW, PW, QW, & TW
 - ...specify vertical variants—V
 - ...specify particular GSUB features—JP90, ljmoXX, vjmoXX & tjmoXX
- A script converts working glyph names to CIDs (*Character IDs*)

- Unicode-based working glyph names can establish unambiguous and unique identity
 - uniXXXX (BMP) or uXXXXX/uXXXXXX (Planes 1 through 16)
- Specify sequences—kana, variation sequences, high-frequency archaic hangul syllables
 - Concatenations of Unicode-based working glyph names
- Use of additional identifiers in Unicode-based working glyph names to...
 - ...specify languages—CN, HK, JP, KR, & TW
 - ...specify glyph width—FW, HW, PW, QW, & TW
 - ...specify vertical variants—V
 - ...specify particular GSUB features—JP90, ljmoXX, vjmoXX & tjmoXX
- A script converts working glyph names to CIDs (*Character IDs*)
- Can be used to insulate developers from CID changes between versions

- Each build can represent a big crank operating on an incredible amount of data



- Each build can represent a big crank operating on an incredible amount of data
 - Source Han Sans / Noto Sans CJK include 458,745 glyphs!

- Each build can represent a big crank operating on an incredible amount of data
 - Source Han Sans / Noto Sans CJK include 458,745 glyphs!
- Possibility of errors is high—*repeatability* & *reliability* are critical!

- Each build can represent a big crank operating on an incredible amount of data
 - Source Han Sans / Noto Sans CJK include 458,745 glyphs!
- Possibility of errors is high—*repeatability* & *reliability are critical!*
- Important for updates

• Compiling OpenType Collections (OTCs) using AFDKO *otf2otc* tool

- Compiling OpenType Collections (OTCs) using AFDKO *otf2otc* tool
 - Saves space by sharing identical 'sfnt' tables across languages & weights

- Compiling OpenType Collections (OTCs) using AFDKO *otf2otc* tool
 - Saves space by sharing identical 'sfnt' tables across languages & weights
 - First release of Source Han Sans / Noto Sans CJK combined **languages** into OTCs

- Compiling OpenType Collections (OTCs) using AFDKO *otf2otc* tool
 - Saves space by sharing identical 'sfnt' tables across languages & weights
 - First release of Source Han Sans / Noto Sans CJK combined languages into OTCs
 - Second & subsequent releases combined **languages & weights** into a single Super OTC



Ken Lunde (小林剣) @ken lunde

Perhaps the **#OpenType** font icon should be changed if it is a Font Collection that has more than 20

fonts. 😴 😻 🌖



7:10 AM · 28 Oct 15

- Compiling OpenType Collections (OTCs) using AFDKO *otf2otc* tool
 - Saves space by sharing identical 'sfnt' tables across languages & weights
 - First release of Source Han Sans / Noto Sans CJK combined **languages** into OTCs
 - Second & subsequent releases combined languages & weights into a single Super OTC
- Copying subroutinized CFF charstring data from one font to another using AFDKO *copyCFFCharstrings* tool

- Compiling OpenType Collections (OTCs) using AFDKO *otf2otc* tool
 - Saves space by sharing identical 'sfnt' tables across languages & weights
 - First release of Source Han Sans / Noto Sans CJK combined **languages** into OTCs
 - Second & subsequent releases combined languages & weights into a single Super OTC
- Copying subroutinized CFF charstring data from one font to another using AFDKO *copyCFFCharstrings* tool
 - Subroutinizing a 65,535-glyph CFF can take several hours



- Compiling OpenType Collections (OTCs) using AFDKO *otf2otc* tool
 - Saves space by sharing identical 'sfnt' tables across languages & weights
 - First release of Source Han Sans / Noto Sans CJK combined **languages** into OTCs
 - Second & subsequent releases combined languages & weights into a single Super OTC
- Copying subroutinized CFF charstring data from one font to another using AFDKO *copyCFFCharstrings* tool
 - Subroutinizing a 65,535-glyph CFF can take several hours
 - Incredibly time-saving when building multiple font resources that differ *only* by default language

Pan-CJK Font Development **Pitfalls**



Pitfall #1 — Completely Trusting Standards

- Standards serve as excellent references—sometimes they are the only reference
 - Do not ignore regional character set and glyph standards—*if available* & *accessible*
- Standards serve as excellent references—sometimes they are the only reference
 - Do not ignore regional character set and glyph standards—*if available* & *accessible*
- Standards are made by humans

- Standards serve as excellent references—sometimes they are the only reference
 - Do not ignore regional character set and glyph standards—*if available* & *accessible*
- Standards are made by humans
- Things made by humans are prone to errors

- Standards serve as excellent references—sometimes they are the only reference
 - Do not ignore regional character set and glyph standards—*if available* & *accessible*
- Standards are made by humans
- Things made by humans are prone to errors
- Thus \rightarrow Standards have the potential to include errors

- Standards serve as excellent references—sometimes they are the only reference
 - Do not ignore regional character set and glyph standards—*if available* & *accessible*
- Standards are made by humans
- Things made by humans are prone to errors
- Thus \rightarrow Standards have the potential to include errors
- If you think that you have discovered an error in a standard, please make the effort to report it!

- Standards serve as excellent references—sometimes they are the only reference
 - Do not ignore regional character set and glyph standards—*if available* & *accessible*
- Standards are made by humans
- Things made by humans are prone to errors
- Thus \rightarrow Standards have the potential to include errors
- If you think that you have discovered an error in a standard, please make the effort to report it!

Your Contact E-mail Address: Your Name:	
Type of Message:	Error Report (Standard, Data files, etc)
Subject:	
	Submit Reset

- Each new version of Unicode adds more characters
 - Unicode Version 8.0 includes more than 120K characters—*over 80K are CJK Unified Ideographs*

- Each new version of Unicode adds more characters
 - Unicode Version 8.0 includes more than 120K characters—*over 80K are CJK Unified Ideographs*

Code Point Range	Block Name	Version	Characters	Characters in Block	Unassigned Code Point	ts
U+3400 U+4DB5	CJK Unified Ideographs Extension A	3.0	6,582		U+4DB6 U+4DBF	10
U+4E00 U+9FA5	CJK Unified Ideographs (URO)	1.1	20,902			
U+9FA6 U+9FBB	CJK Unified Ideographs (URO)	4.1	22			
U+9FBC U+9FC3	CJK Unified Ideographs (URO)	5.1	8	20,950 U+9FD6 U+9FFF		42
U+9FC4 U+9FCB	CJK Unified Ideographs (URO)	5.2	8			
U+9FCC	CJK Unified Ideographs (URO)	6.1	1			
U+9FCD U+9FD5	CJK Unified Ideographs (URO)	8.0	9			
U+FA0E U+FA0F U+FA11 U+FA13 U+FA14 U+FA1F U+FA21 U+FA23 U+FA24 U+FA27 U+FA29	CJK Compatibility Ideographs	1.1	12		N/A	
U+20000 U+2A6D6	CJK Unified Ideographs Extension B	3.1	42,711		U+2A6D7 U+2A6DF	9
U+2A700 U+2B734	CJK Unified Ideographs Extension C	5.2	4,149		U+2B735 U+2B73F	11
U+2B740 U+2B81D	CJK Unified Ideographs Extension D	6.0	222		U+2B81E U+2B81F	2
U+2B820 U+2CEA1	CJK Unified Ideographs Extension E	8.0	5,762		U+2CEA2 U+2CEAF	14
TOTAL			80,388			88

- Each new version of Unicode adds more characters
 - Unicode Version 8.0 includes more than 120K characters—*over 80K are CJK Unified Ideographs*
- Regional standards may change in various ways
 - Representative glyph changes—JIS90 versus JIS2004 for JIS X 0208 & JIS X 0213 (Japan)
 - Additional characters—JIS X 0213:2000 versus JIS X 0213:2004 (Japan)
 - New standards—*HKCS* (*Hong Kong Character Set*) 2015

- Each new version of Unicode adds more characters
 - Unicode Version 8.0 includes more than 120K characters—*over 80K are CJK Unified Ideographs*
- Regional standards may change in various ways
 - Representative glyph changes—JIS90 versus JIS2004 for JIS X 0208 & JIS X 0213 (Japan)
 - Additional characters—JIS X 0213:2000 versus JIS X 0213:2004 (Japan)
 - New standards—HKCS (Hong Kong Character Set) 2015
- Easily overlooked regional standards
 - China's TGH 2013 (通用规范汉字表)

- Each new version of Unicode adds more characters
 - Unicode Version 8.0 includes more than 120K characters—*over 80K are CJK Unified Ideographs*
- Regional standards may change in various ways
 - Representative glyph changes—JIS90 versus JIS2004 for JIS X 0208 & JIS X 0213 (Japan)
 - Additional characters—JIS X 0213:2000 versus JIS X 0213:2004 (Japan)
 - New standards—*HKCS* (*Hong Kong Character Set*) 2015
- Easily overlooked regional standards
 - China's TGH 2013 (通用规范汉字表)





- Taiwan (ROC) & Hong Kong SAR have different regional glyph conventions/standards

- Taiwan (ROC) & Hong Kong SAR have different regional glyph conventions/standards
- TW follows the Taiwan MOE (*Ministry of Education*) glyph standards

- Taiwan (ROC) & Hong Kong SAR have different regional glyph conventions/standards
- TW follows the Taiwan MOE (*Ministry of Education*) glyph standards



- Taiwan (ROC) & Hong Kong SAR have different regional glyph conventions/standards
- TW follows the Taiwan MOE (*Ministry of Education*) glyph standards
- HK is somewhat of a hybrid

- Taiwan (ROC) & Hong Kong SAR have different regional glyph conventions/standards
- TW follows the Taiwan MOE (*Ministry of Education*) glyph standards
- HK is somewhat of a hybrid



- Taiwan (ROC) & Hong Kong SAR have different regional glyph conventions/standards
- TW follows the Taiwan MOE (*Ministry of Education*) glyph standards
- HK is somewhat of a hybrid
 - Some glyphs are identical to TW, CN, or JP forms
 - Some glyphs require a form that is unique to HK

- Taiwan (ROC) & Hong Kong SAR have different regional glyph conventions/standards
- TW follows the Taiwan MOE (*Ministry of Education*) glyph standards
- HK is somewhat of a hybrid
 - Some glyphs are identical to TW, CN, or JP forms
 - Some glyphs require a form that is unique to HK



- Taiwan (ROC) & Hong Kong SAR have different regional glyph conventions/standards
- TW follows the Taiwan MOE (*Ministry of Education*) glyph standards
- HK is somewhat of a hybrid
 - Some glyphs are identical to TW, CN, or JP forms
 - Some glyphs require a form that is unique to HK
- HKCS (*Hong Kong Character Set*) 2015 is expected to replace HKSCS-2008

- Building OpenType fonts that include language-specific features is relatively easy
 - This is mainly for the 'locl' (*Localized Forms*) GSUB feature, but also affects 'vert' (*Vertical Alternates*)

- Building OpenType fonts that include language-specific features is relatively easy
 - This is mainly for the 'locl' (*Localized Forms*) GSUB feature, but also affects 'vert' (*Vertical Alternates*)
- Unfortunately, very few applications support language-tagging

- Building OpenType fonts that include language-specific features is relatively easy
 - This is mainly for the 'locl' (*Localized Forms*) GSUB feature, but also affects 'vert' (*Vertical Alternates*)
- Unfortunately, very few applications support language-tagging
 - Some modern browsers—*Chrome & Firefox are good examples*



- Building OpenType fonts that include language-specific features is relatively easy
 - This is mainly for the 'locl' (*Localized Forms*) GSUB feature, but also affects 'vert' (*Vertical Alternates*)
- Unfortunately, very few applications support language-tagging
 - Some modern browsers—*Chrome & Firefox are good examples*
 - Adobe InDesign



- Building OpenType fonts that include language-specific features is relatively easy
 - This is mainly for the 'locl' (*Localized Forms*) GSUB feature, but also affects 'vert' (*Vertical Alternates*)
- Unfortunately, very few applications support language-tagging
 - Some modern browsers—*Chrome & Firefox are good examples*
 - Adobe InDesign
- Some applications support language tagging at the document level via preferences
 - Unexpected results → Install font with Simplified Chinese default but with Traditional Chinese preferences

• Some punctuation & symbols require language- or region-specific forms



- Some punctuation & symbols require language- or region-specific forms
 - This can be a *no-win scenario* for some characters—*single* & *double smart quotes*

- Some punctuation & symbols require language- or region-specific forms
 - This can be a *no-win scenario* for some characters—*single* & *double smart quotes*

	U+2018	U+2019	U+201C	U+201D
Full-width	-' 6 '-	-'9 '-	-' 66 ^{'-}	- ' ?? '-
	n c	n e	n e	n c
Proportional	_' '_ 6	-' <u>'</u> -)	_' '_ {	-')) '-
	n e	n c	n e	n e

- Some punctuation & symbols require language- or region-specific forms
 - This can be a *no-win scenario* for some characters—*single* & *double smart quotes*

	U+2018	U+2019	U+201C	U+201D
Full-width	_' <u></u>	-' ? '-	-' 66 '-	-' ?? '-
	n e	n c	n c	n e
Proportional	_! !_ 6	-' '-)	_ا _ا_ د د	_' >> '
	n e	n e	n e	т. с.
Proportional CJK	-' ' '-	-''' -	-' ‹‹ '-	-' >> '-
	n c	ъ. с.		

- Some punctuation & symbols require language- or region-specific forms
 - This can be a no-win scenario for some characters—single & double smart quotes
- Vertical forms
 - Do not depend on vertical forms that happen to be encoded—*see U+FEXX*

- Some punctuation & symbols require language- or region-specific forms
 - This can be a *no-win scenario* for some characters—*single* & *double smart quotes*
- Vertical forms
 - Do not depend on vertical forms that happen to be encoded—*see U+FEXX*
- Half-width ASCII

- Some punctuation & symbols require language- or region-specific forms
 - This can be a *no-win scenario* for some characters—*single* & *double smart quotes*
- Vertical forms
 - Do not depend on vertical forms that happen to be encoded—see U+FEXX
- Half-width ASCII
- Characters that are supported via sequences

Deployment format considerations

- Deployment format considerations
 - Single- or multiple-language default?

- Deployment format considerations
 - Single- or multiple-language default?
 - Individual font resources or font collections?

- Deployment format considerations
 - Single- or multiple-language default?
 - Individual font resources or font collections?
- Language-switching via 'locl' GSUB feature & language tagging? → single- or multiple-language default

- Deployment format considerations
 - Single- or multiple-language default?
 - Individual font resources or font collections?
- Language-switching via 'locl' GSUB feature & language tagging? → single- or multiple-language default
- Language-switching via separate font resources? → *multiple-language default*

- Deployment format considerations
 - Single- or multiple-language default?
 - Individual font resources or font collections?
- Language-switching via 'locl' GSUB feature & language tagging? → single- or multiple-language default
- Language-switching via separate font resources? → *multiple-language default*
- Do both? → Very Good Idea[™]
• The architectural limit for the number of glyphs in a 'glyf' (TrueType) table or CFF is 65,535

- The architectural limit for the number of glyphs in a 'glyf' (TrueType) table or CFF is 65,535
- The number of glyphs required for a Pan-CJK font depends on several important factors

- The architectural limit for the number of glyphs in a 'glyf' (TrueType) table or CFF is 65,535
- The number of glyphs required for a Pan-CJK font depends on several important factors
 - Character set scope
 - Supported languages—this influences the number of glyphs per code point
 - Typeface style—such as serif versus sans serif versus script
 - Typeface design

- The architectural limit for the number of glyphs in a 'glyf' (TrueType) table or CFF is 65,535
- The number of glyphs required for a Pan-CJK font depends on several important factors
 - Character set scope
 - Supported languages—this influences the number of glyphs per code point
 - Typeface style—such as serif versus sans serif versus script
 - Typeface design
- What happens if one exceeds the 64K glyph limit?

- The architectural limit for the number of glyphs in a 'glyf' (TrueType) table or CFF is 65,535
- The number of glyphs required for a Pan-CJK font depends on several important factors
 - Character set scope
 - Supported languages—this influences the number of glyphs per code point
 - Typeface style—such as serif versus sans serif versus script
 - Typeface design
- What happens if one exceeds the 64K glyph limit?
 - Use separate font resources for large/independent scripts—*Korean hangul letters, syllables* & symbols

S What Else?

Source Han Sans ExtraLight — 65,535 glyphs — Noto Sans CJK Thin

Image: State		//////////////////////////////////////	SI 44 4 111 5 50 000 × 0000 0000000000000000000	00000000000000000000000000000000000000		калаана баваба киноониян сайла учунун артаа кала кала кала •••••••••••••••••••••••••••••••••••	
				andra an Angre andra and	andread and a second		and a second sec
	Antonio						
中が加払加になったことにあったして、おりかなどの利用した。 のものは、2000年に、2000	anta anta anta anta anta anta anta anta						
18 使用此的问题。如果我们的意思,我们就是我们的问题。我们就是我们的问题。 我们就是我们的问题。我们就是我们的问题。我们就是我们的问题。 我们就是我们的问题。我们就是我们的问题。 我们就是我们的问题。我们就是我们的问题。我们就是我们的问题。 我们就是我们的问题。我们就是我们的问题。我们就是我们的问题。 我们就是我们的问题。我们就是我们的我们就是我们的我们的我们就是我们的我们的。 我们就是我们的我们的我们就是我们就是我们的我们就是我们的我们就是我们的我们的我们。 我们就是我们的我们就是我们就是我们的我们就是我们的我们就是我们的我们的我们就是我们的我们的我们就是我们就是我们的我们的我们就是我们的我们的我们就是我们的我们的我们就是我们的我们的我们就是我们就是我们的我们就是我们的我们就是我们的我们就是我们的我们就是我们就是我们的我们就是我们的我们就是我们的我们就是我们的我们就是我们的我们就是我们我们就是我们的我们就是我们就是我们我们我们我们就是我们我们就是我们我们我们我们我们就是我们我们我们我们							
						For the second	
第二次等温率通常定置目標準備について、日本市が、市市が、市市が、市市、市市、市市、市市、市市、市市、市市、市市、市市、市市			an a				
			 コートレーン・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	コールイルテービース・シーム・シーム・シーム・シーム・シーム・シーム・シーム・シーム・シーム・シーム	ーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーー	1.4月~~4月~~4月~4月~4日~4日~4日~4日~11~11~11~11~11~11~4月~4月~4日4日~4月~4日 1.4月~~4日~4日~4日~4日~4日~4日~5日~5日~5日~5日~5日~5日~5日~5日~5日~5日~5日~5日~5日	$ \frac{1}{2} \left[1 \right] \frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2} \frac{1}{2} - \frac{1}{2} \frac$

© 2015 Adobe Systems Incorporated. All Rights Reserved. IUC39—Santa Clara. CA. USA. Earth

I Love It When A Plan Comes Together

More Information

- Developing & Deploying The World's First Open Source Pan-CJK Typeface Family (IUC38 presentation) <u>http://blogs.adobe.com/CCJKType/files/2014/11/iuc38-lunde-s1t3.pdf</u>
- Building Source Han Sans & Noto Sans CJK (IUC38 presentation) <u>http://blogs.adobe.com/CCJKType/files/2014/11/iuc38-lunde-s8t1.pdf</u>
- CJK Type Blog <u>http://blogs.adobe.com/CCJKType/</u>
- Taiwan MOE Glyph Standards

http://www.edu.tw/pages/list.aspx?Node=3691&Index=7&wid=c5ad5187-55ef-4811-8219-e946fe04f725

Unicode Contact Form—for reporting errors

http://www.unicode.org/reporting.html



