

$$\left(\sum_{k=1}^n a_k b_k \right)^2 \leq \left(\sum_{k=1}^n a_k^2 \right) \left(\sum_{k=1}^n b_k^2 \right)$$

Where Fonts Have Been...

Where Fonts Have Been...

Evolution of TrueType

- Apple format
- Container format, set of known data tables
- Scalable glyphs with quadratic splines
- Character mapping table, metrics, kerning and hinting data
- Classic TrueType font (.ttf)
- Apple shares with Microsoft

TrueType GX (1994)

- Apple extends TrueType
- i.e. adds tables (e.g. mort)
- Adds support for richer typographical variations
- Allows handling of complex scripts
- Apple doesn't share

OpenType

- Microsoft creates TrueType Open
- i.e. adds tables (e.g. GPOS, GDEF)
- Together with Adobe creates OpenType
- Adds Postscript style glyph format table
- Two glyph formats - quadratic (.ttf) and cubic (.otf)
- Apple changes TrueType GX name to AAT

Problems with TrueType format

- Dual platform nature
- Family names are per-platform (!!!)
- Ways of classifying fonts differs
- Weight pulled from OS/2 table (Microsoft)
- Weight based on style name (Apple)
- Microsoft defines embedding bits, Apple ignores
- Complex script handling completely different

TrueType name table

- Family, subfamily, full, version, license
- For each name, two platforms, n localizations
- Different style names for each weight

Style linking

- Unifying a set of faces into a single font family
- Apple spec: family name applies across all style variations
- Microsoft spec: restrict family name, add extra family name
- Family name - only regular, bold, italic, bold italic
- Preferred Family name - used to group larger set
- "Savvy" apps: Preferred Family ➔ Family

Name example

Macintosh: 1 family, 6 weights
Kozuka Gothic Pro (extra light, light, regular, medium,
bold, heavy)

Microsoft: 6 families, 1 weight each
Kozuka Gothic Pro EL
Kozuka Gothic Pro L
Kozuka Gothic Pro R
Kozuka Gothic Pro M
Kozuka Gothic Pro B
Kozuka Gothic Pro H

but "savvy" apps can use the Preferred Family name

Implications?

- Browsers need to become "savvy" apps on Windows
- How to map "old-style" family names?
- Luckily very few in actual use
- Spec bugs hidden by lack of style-linked families

More information

- Windows, CSS font families & OpenType1.5
- <http://blogs.adobe.com/typblography/atypi2006/CSS%20&%20OT%2015.pdf>
- WPF font selection model
- <http://blogs.msdn.com/text/attachment/2249036.ashx>
- Check out Apple Font Tools, ttx tool for dumping font data!

CSS 2.1 font matching

- Match family name to a set of font faces
- Match font-style (italic/oblique), fallback on failure
- Match font-variant, fallback if not synthesized (archaic)
- Match font-weight, never fails
- For bitmap fonts, match font size

CSS 2.1 font matching

- Effectively, font-style trumps font-family
- Reality is most user agents either synthesize or ignore
- Many scripts have no concept of italics
- CSS3: no fallback on italic, no matching of font-variant
- italic/oblique equivalent *unless* both faces exist
- Doesn't break compatibility, just reflects actual behavior

The Road From Here...

The Road From Here...

Simplified web fonts definition

- Trimmed down version of CSS2 @font-face spec
- font-family, src url, src local (with format hint)
- font attribute descriptors (font-weight, font-style, etc.)
- No attribute-based font substitution
- Safari already implements, Mozilla, Opera coming soon

Motivations

- Richer set of available fonts
- Much tighter control over how content is rendered
- Better support for international sites (e.g. BBC)
- Domains with specialized font requirements (e.g. MathML)
- Improved integration of SVG fonts in HTML content
- Need for restricted font formats is a separate discussion

Examples

Subtleties

- Do descriptors define underlying attributes?
- CSS2 descriptors feel like hints
- Probably not very intuitive for web authors
- Use default values instead?
- Font matching based on descriptors, not underlying font info

Descriptor as a Hint

```
@font-face {
  font-family: Bongo;
  src: url(Bongo.ttf);
}

@font-face {
  font-family: Bongo;
  src: url(BongoItalic.ttf);
  font-style: italic;
}
```

```
body { font-family: Bongo; font-weight: bold; }
```

Use This Font!

```
@font-face {  
  font-family: Bongo;  
  src: url(BongoNarrowLightItalic.ttf);  
}
```

```
body { font-family: Bongo; }
```

Security Issues

- Need to be very careful about handling fonts
- Intentionally corrupted fonts become an attack vector
- Errors can occur with metrics, at drawing time, ...
- Some validation possible but not a silver bullet
- Fonts associated with page A should not affect page B
- Seems obvious, but font APIs make this a bit tricky
- Don't use downloaded fonts in system fallback!

Thank You Webkit!

```
// The value "3" means that the font is private  
// and can't be seen by anyone else.
```

```
ATSTFontActivateFromMemory((void*)buffer->data(),  
  buffer->size(),  
  3,  
  kATSTFontFormatUnspecified,  
  NULL,  
  kATSTOptionFlagsDefault,  
  &containerRef);
```

Beyond Web Fonts...

Beyond Web Fonts...

Size Adjustments

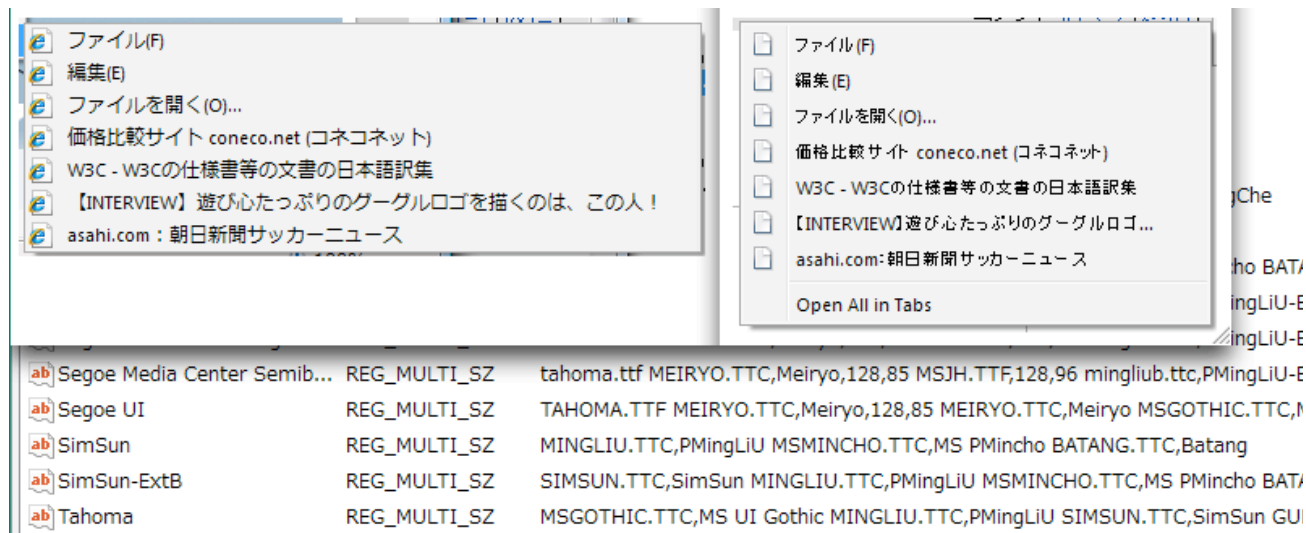
- CSS2 font-size-adjust
- Tried to improve the explanation
- No list of aspect values!
- Example of how to calculate
- Implemented in Firefox 3!

Beyond x-height adjustments

- Font linking on CJK Windows
- Tahoma is a magic font that links to other CJK fonts
- Vista: Segoe UI

- On fallback to CJK font, resize occurs
- Hmmm, add font-size-adjust setting?

Font Link Size Adjustments

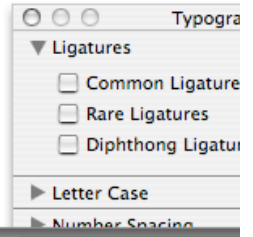


Richer Typography

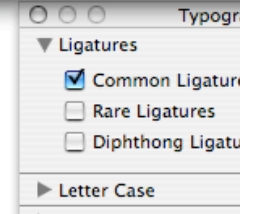
- Exposing OpenType features
- Extend font-variant?
- Ligatures, figure forms, lots of language specific figures
- Håkon's proposal:
- <http://lists.w3.org/Archives/Public/www-style/2008Jan/0380.html>

Ligatures

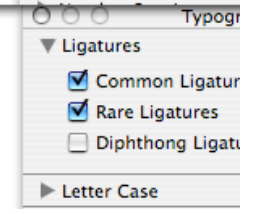
final stand



final stand



final stand

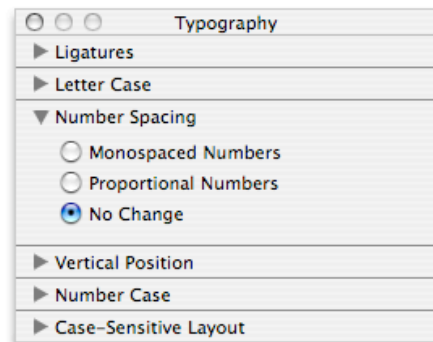


Number Forms

Final totals:

\$9,389,754

\$6,111,111

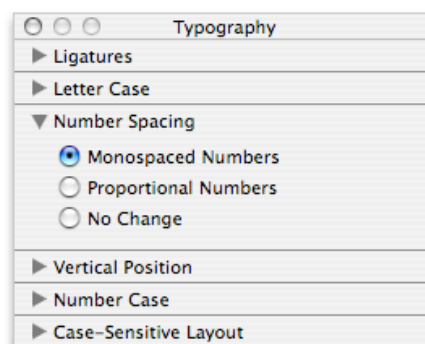


Number Forms

Final totals:

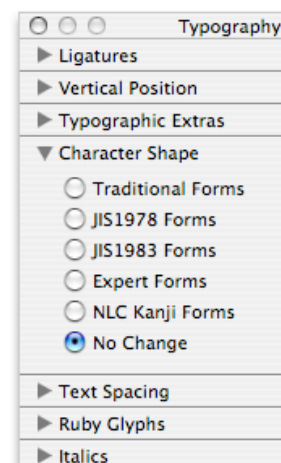
\$9,389,754

\$6,111,111



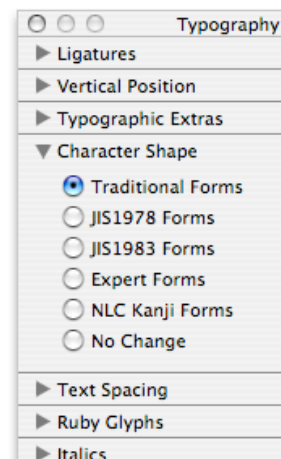
Alternative Glyphs

大学



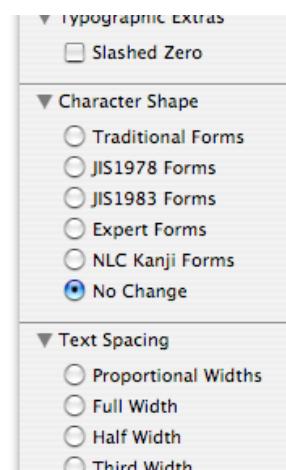
Alternative Glyphs

大學



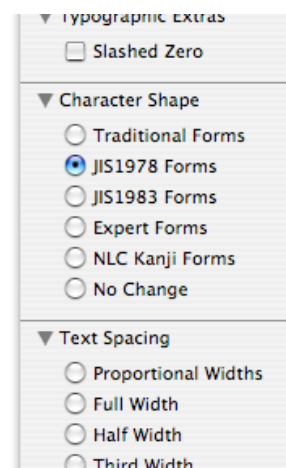
Alternative Glyphs

麴町



Alternative Glyphs

麥町



Font Rendering Properties

- Safari: -webkit- text properties
- text-fill-color, text-stroke-color, text-stroke-width
- Interaction with SVG fill/stroke properties?

Lastly...

おしまい
m(_)m