



Setting  
standards-  
friendly  
web type

# Setting standards-friendly *web* type

❖ The art of technological typography

---

Simon Pascal Klein – @klepas

I'm...

I'm...

1. one 'of those' BA students;

I'm...

1. one 'of those' BA students;
2. a graphic & web designer;

I'm...

1. one 'of those' BA students;
2. a graphic & web designer;
3. and, incurably, a rampant 'typophile'.

‘Typophiles’ are über geeks who  
♥ typography.

‘Typophiles’ are über geeks who  
♥ typography.

Hands up and make yourselves known.

ty•pog•ra•phy | tī'pägrəfē | •n

# ty•pog•ra•phy | tī'pägrəfē | •n

Typography is the art of creating and setting type  
with the purpose of honouring the text it sets.



1  
**Getting a typeface  
on the web.**

2

1  
**Getting a typeface  
on the web.**

2  
**Setting that type  
all pretty.**

1



Q: Why?



*premise:*

*premise:*

The internet is saturated;  
it's info-heavy.

*premise:*

The internet is saturated;  
it's info-heavy.

Much of it is textual.

*premise:*

The internet is saturated;  
it's info-heavy.

Much of it is textual.

Much of it is a pain to read.



Typography can bring aesthetic order to information, helping users read & navigate.



# Renewed interest in Typography

# Renewed interest in Typography

- central to aesthetics;

# Renewed interest in Typography

- central to aesthetics;
- central to accessibility, legibility, ...;

# Renewed interest in Typography

- central to aesthetics;
- central to accessibility, legibility, ...;
- many finer typesetting techniques are ‘new’ and admired for their ingenuity;

# Renewed interest in Typography

- central to aesthetics;
- central to accessibility, legibility, ...;
- many finer typesetting techniques are ‘new’ and admired for their ingenuity;
- growing availability of web fonts.



i.e. make textual stuff look pretty.







Web experiences that feature good typography  
are easier and more pleasurable to use.



# Assumptions?

# Assumptions?

It sucks? It's boring?

# Assumptions?

It sucks? It's boring?

1. Limited to sans-serif only for readability?

# Assumptions?

It sucks? It's boring?

1. Limited to sans-serif only for readability?
2. Only ‘web-safe’ fonts?

# Assumptions?

It sucks? It's boring?

1. Limited to sans-serif only for readability?
2. Only ‘web-safe’ fonts?
3. Limited control—gotta resort to Flash and other non-standard technologies?

# Assumptions?

It sucks? It's boring?

1. Limited to sans-serif only for readability?
2. Only ‘web-safe’ fonts?
3. Limited control—gotta resort to Flash and other non-standard technologies?
4. So many options; how do I *do* it?



Not quite.

Not quite.

(Yay!)



# Possible problems & pitfalls

# Possible problems & pitfalls

- more fonts ≠ instant good typography;

## Possible problems & pitfalls

- more fonts ≠ instant good typography;
- many of the new web fonts aren't optimised or even designed for web use;

## Possible problems & pitfalls

- more fonts ≠ instant good typography;
- many of the new web fonts aren't optimised or even designed for web use;
- growing availability of web fonts;

## Possible problems & pitfalls

- more fonts ≠ instant good typography;
- many of the new web fonts aren't optimised or even designed for web use;
- growing availability of web fonts;
- differences in implementation;

# Possible problems & pitfalls

- more fonts ≠ instant good typography;
- many of the new web fonts aren't optimised or even designed for web use;
- growing availability of web fonts;
- differences in implementation;
- closed vs. open implementation struggle;

# Possible problems & pitfalls

- more fonts ≠ instant good typography;
- many of the new web fonts aren't optimised or even designed for web use;
- growing availability of web fonts;
- differences in implementation;
- closed vs. open implementation struggle;
- sub-setting and compression becomes your responsibility if not using a hosting service.



Q: How?

A: There are a range of typesetting options for the web that vary in technical complexity and stylistic versatility.

A: There are a range of typesetting options for the web that vary in technical complexity and stylistic versatility.

They're standards-based + you can use them now.

## Options for setting fonts:

Options for setting fonts:

1. Installed fonts (*mostly* web-safe)

## Options for setting fonts:

1. Installed fonts (*mostly* web-safe)
2. SIFR or other JS and Flash replacement

## Options for setting fonts:

1. Installed fonts (*mostly* web-safe)
2. SIFR or other JS and Flash replacement
3. *Cufón et al.* (<http://tinyurl.com/git-cufon>)

## Options for setting fonts:

1. Installed fonts (*mostly* web-safe)
2. SIFR or other JS and Flash replacement
3. *Cufón* et al. (<http://tinyurl.com/git-cufon>)
4. webfonts: EOT/EOT Lite via @font-face

## Options for setting fonts:

1. Installed fonts (*mostly* web-safe)
2. SIFR or other JS and Flash replacement
3. *Cufón* et al. (<http://tinyurl.com/git-cufon>)
4. webfonts: EOT/EOT Lite via @font-face
5. webfonts: OT/TTF via @font-face

## Options for setting fonts:

1. Installed fonts (*mostly* web-safe)
2. SIFR or other JS and Flash replacement
3. *Cufón* et al. (<http://tinyurl.com/git-cufon>)
4. webfonts: EOT/EOT Lite via @font-face
5. webfonts: OT/TTF via @font-face
6. webfonts: WOFF via @font-face

## Options for setting fonts:

1. Installed fonts (*mostly* web-safe)
2. SIFR or other JS and Flash replacement
3. *Cufón* et al. (<http://tinyurl.com/git-cufon>)
4. webfonts: EOT/EOT Lite via @font-face
5. webfonts: OT/TTF via @font-face
6. webfonts: WOFF via @font-face
7. webfonts: SVG via @font-face

## Options for setting fonts:

1. Installed fonts (*mostly* web-safe)
2. SIFR or other JS and Flash replacement
3. *Cufón* et al. (<http://tinyurl.com/git-cufon>)
4. webfonts: EOT/EOT Lite via @font-face
5. webfonts: OT/TTF via @font-face
6. webfonts: WOFF via @font-face
7. webfonts: SVG via @font-face
8. webfonts: hosting & licensing services

# 1. Installed or web-safe fonts

## 1. Installed or web-safe fonts

Simply call upon them directly in your font stack declarations, e.g.:

## 1. Installed or web-safe fonts

Simply call upon them directly in your font stack declarations, e.g.:

```
font-family: Baskerville,  
             Times,  
             'Times New Roman',  
             serif;
```

# 1. Installed or web-safe fonts

Simply call upon them directly in your font stack declarations, e.g.:

```
font-family: Baskerville,  
             Times,  
             'Times New Roman',  
             serif;
```

```
font-family: 'Helvetica Neue',  
             Helvetica,  
             'Arial',  
             sans-serif;
```



## Font stacks:

## Font stacks:

- desired

## Font stacks:

- desired
- fallback

## Font stacks:

- desired
- fallback
- generic (e.g. serif, sans-serif, monospace)

## 2. SIFR et al.

Flash text replacement that in unison with js  
replaces type using type from a Flash file.

## 2. SIFR et al.

Flash text replacement that in unison with js  
replaces type using type from a Flash file.

Not a feasible solution long-term (i.e. don't use):

## 2. SIFR et al.

Flash text replacement that in unison with js  
replaces type using type from a Flash file.

Not a feasible solution long-term (i.e. don't use):

1. does subsetting;

## 2. SIFR et al.

Flash text replacement that in unison with js  
replaces type using type from a Flash file.

Not a feasible solution long-term (i.e. don't use):

1. does subsetting;
2. non-standard: relies on both Flash and JS;

## 2. SIFR et al.

Flash text replacement that in unison with js  
replaces type using type from a Flash file.

Not a feasible solution long-term (i.e. don't use):

1. does subsetting;
2. non-standard: relies on both Flash and JS;
3. performance: high page load time—not feasible  
for body copy.

### 3. Cufón & JS implementations

These were the first js-only implementations.

Cufón converts font paths to VML paths stored in JSON, then rendering the font paths using JS.

### 3. Cufón & JS implementations

These were the first js-only implementations.

Cufón converts font paths to VML paths stored in JSON, then rendering the font paths using JS.

Thoughts and concerns:

### 3. Cufón & JS implementations

These were the first js-only implementations.

Cufón converts font paths to VML paths stored in JSON, then rendering the font paths using JS.

Thoughts and concerns:

1. not accessible but works in most browsers;

### 3. Cufón & JS implementations

These were the first js-only implementations.

Cufón converts font paths to VML paths stored in JSON, then rendering the font paths using JS.

Thoughts and concerns:

1. not accessible but works in most browsers;
2. type foundries are sceptical—not a ‘service’;  
limited palette of typefaces;

### 3. Cufón & JS implementations

These were the first js-only implementations.

Cufón converts font paths to VML paths stored in JSON, then rendering the font paths using JS.

Thoughts and concerns:

1. not accessible but works in most browsers;
2. type foundries are sceptical—not a ‘service’;  
limited palette of typefaces;
3. converter is often proprietary.

## 4. webfonts: EOT/EOT Lite

Developed by Microsoft, Embedded OpenType subsets, compresses, and encrypts OT and TTF. Can be restricted a list of trusted roots. Uses webfonts. Described as ‘DRM icing on an OpenType cake’.

## 4. webfonts: EOT/EOT Lite

Developed by Microsoft, Embedded OpenType subsets, compresses, and encrypts OT and TTF. Can be restricted a list of trusted roots. Uses webfonts. Described as ‘DRM icing on an OpenType cake’.

Not feasible as-is currently:

## 4. webfonts: EOT/EOT Lite

Developed by Microsoft, Embedded OpenType subsets, compresses, and encrypts OT and TTF. Can be restricted a list of trusted roots. Uses webfonts. Described as ‘DRM icing on an OpenType cake’.

Not feasible as-is currently:

1. non-standard, proprietary solution;

## 4. webfonts: EOT/EOT Lite

Developed by Microsoft, Embedded OpenType subsets, compresses, and encrypts OT and TTF. Can be restricted a list of trusted roots. Uses webfonts. Described as ‘DRM icing on an OpenType cake’.

Not feasible as-is currently:

1. non-standard, proprietary solution;
2. requires a proprietary converter (WEFT) to use;

## 4. webfonts: EOT/EOT Lite

Developed by Microsoft, Embedded OpenType subsets, compresses, and encrypts OT and TTF. Can be restricted a list of trusted roots. Uses webfonts. Described as ‘DRM icing on an OpenType cake’.

Not feasible as-is currently:

1. non-standard, proprietary solution;
2. requires a proprietary converter (WEFT) to use;
3. proposal rejected by w3c & **only** supported by IE.

## 5. webfonts: OT/TTF

Also uses webfonts css `@font-face` and links directly via the `src:` declaration to an OT/TTF file which is downloaded and used to render the type.

## 5. webfonts: OT/TTF

Also uses webfonts css `@font-face` and links directly via the `src:` declaration to an OT/TTF file which is downloaded and used to render the type.

Feasible...:

## 5. webfonts: OT/TTF

Also uses webfonts css `@font-face` and links directly via the `src:` declaration to an OT/TTF file which is downloaded and used to render the type.

Feasible...:

1. OT/TTF files can easily be downloaded;

## 5. webfonts: OT/TTF

Also uses webfonts css `@font-face` and links directly via the `src:` declaration to an OT/TTF file which is downloaded and used to render the type.

Feasible...:

1. OT/TTF files can easily be downloaded;
2. support: Safari 3.1+, Firefox 3.5+, Opera 10+, Chrome 4+; IE9 dev, Android 2.2+;

## 5. webfonts: OT/TTF

Also uses webfonts css `@font-face` and links directly via the `src:` declaration to an OT/TTF file which is downloaded and used to render the type.

Feasible...:

1. OT/TTF files can easily be downloaded;
2. support: Safari 3.1+, Firefox 3.5+, Opera 10+, Chrome 4+; IE9 dev, Android 2.2+;
3. subsetting & compression comes down to author.

# The Potential of Web Typography:

@FONT-FACE AND FIREFOX 3.5

by Ian Lynam & Craig Mod

  
FIREFOX 3.5 IS OUT. AND THE MORE USERS DOWNLOAD IT, THE more designers will be able to take advantage of the @font-face css rule. How can @font-face be used with currently implemented css selectors to create engaging, nuanced and more mature typography? Let's find out.

## @font-face — what it is exactly?

@FONT-FACE IS A CSS RULE IMPLEMENTED IN Firefox's latest 3.5 browser release. It allows web designers to reference fonts not installed on end user machines.

(NOTE: John Daggett has written [a lovely primer article on @font-face](#). It serves as a fine reference into the nitty gritty of @font-face implementation and usage.)

Fine typography has always been one of the stumbling points of web design. Limited at most to a handful of cross-platform specific fonts, web designers have often found it necessary to hack their way into typographically nuanced territory. Technologies such as [sIFR](#) and [Cufon](#) utilize Flash and Javascript techniques to achieve typographic variety, but often at the expense of clumsy technological implementation. @font-face brings with it the hope of a standard, cross-platform, cross-browser, lightweight method for referencing font-files not found on end users' computer. In other words: it allows web designers to store fonts on their server and reference them in css, regardless of what fonts the user browsing the page has installed.

## What we'd like to see added

HERE ARE SOME TYPOGRAPHIC CONTROLS WE'D love to see implemented in the css specifications:

### Baseline-shift

It's sometimes necessary to [nudge the baseline](#) of certain glyphs in a line. We'd like to have control over this as a css attribute and to be able apply it selectively through span tags and classes.

### Hyphenation

Having granular control over [hyphenation would instantly and drastically change](#) the perceived visual texture of online typography. (You can read about the current state of the hyphenation specification on the [w3c homepage](#).)

### Numerical selection

Most contemporary typefaces are offered with both [lining](#) and [non-lining numerals](#). Having access to these alternate characters would be great.

And more wistfully ...

We'd like to see more control over OpenType-specific

## Resources

### GENERAL WEB TYPOGRAPHY

- [The Elements of Typographic Style Applied to the Web](#)
- [Deciphering Baseline Rhythm](#)
- [Setting Web Typography to a Baseline Grid](#)
- [Open Source Font Library](#)

### GRID SYSTEMS

- [The Grid System](#)

### @FONT-FACE FONT RESOURCES

- [Exlibris Font Foundry](#)
- [A wiki listing fonts and foundries offering @font-face compatible EULAs](#)

### @FONT-FACE DOCS

- [John Daggett's @font-face primer](#)
- [MDC @font-face documentation](#)
- [css2 Fonts specification](#)
- [css3 Fonts draft](#)
- [Cross-Origin Resource Sharing](#)

## 6. webfonts: WOFF

Works just as OT/TTF, linking directly via the `src`:  
declaration to a WOFF file (compressed font data +  
XML meta-data bundle, built using `sfnt2woff`).

## 6. webfonts: WOFF

Works just as OT/TTF, linking directly via the `src`: declaration to a WOFF file (compressed font data + XML meta-data bundle, built using `sfnt2woff`).

Thoughts (besides being awesome):

## 6. webfonts: WOFF

Works just as OT/TTF, linking directly via the `src`: declaration to a WOFF file (compressed font data + XML meta-data bundle, built using `sfnt2woff`).

Thoughts (besides being awesome):

1. CORS available via HTTP response headers;

## 6. webfonts: WOFF

Works just as OT/TTF, linking directly via the `src`: declaration to a WOFF file (compressed font data + XML meta-data bundle, built using `sfnt2woff`).

Thoughts (besides being awesome):

1. CORS available via HTTP response headers;
2. the sfnt-font data can be extracted... effort?;

## 6. webfonts: WOFF

Works just as OT/TTF, linking directly via the `src`: declaration to a WOFF file (compressed font data + XML meta-data bundle, built using `sfnt2woff`).

Thoughts (besides being awesome):

1. CORS available via HTTP response headers;
2. the sfnt-font data can be extracted... effort?;
3. support: Firefox 3.6+, WebKit, Chrome 5+, IE9 dev;

## 6. webfonts: WOFF

Works just as OT/TTF, linking directly via the `src`: declaration to a WOFF file (compressed font data + XML meta-data bundle, built using `sfnt2woff`).

Thoughts (besides being awesome):

1. CORS available via HTTP response headers;
2. the sfnt-font data can be extracted... effort?;
3. support: Firefox 3.6+, WebKit, Chrome 5+, IE9 dev;
4. subsetting & compression comes down to author.

## 7. webfonts: SVG

Works just as EOT and OT/TTF, instead linking directly via the `src:` declaration to an SVG file which is downloaded and used to render the type.

## 7. webfonts: SVG

Works just as EOT and OT/TTF, instead linking directly via the `src:` declaration to an SVG file which is downloaded and used to render the type.

Thoughts:

## 7. webfonts: SVG

Works just as EOT and OT/TTF, instead linking directly via the `src:` declaration to an SVG file which is downloaded and used to render the type.

Thoughts:

1. again, SVG files can easily be downloaded;

## 7. webfonts: SVG

Works just as EOT and OT/TTF, instead linking directly via the `src:` declaration to an SVG file which is downloaded and used to render the type.

Thoughts:

1. again, SVG files can easily be downloaded;
2. support: Firefox 3.5+ Chrome 0.3+, Opera 9+, iOS1+, Safari 3.1+;

## 7. webfonts: SVG

Works just as EOT and OT/TTF, instead linking directly via the `src:` declaration to an SVG file which is downloaded and used to render the type.

Thoughts:

1. again, SVG files can easily be downloaded;
2. support: Firefox 3.5+ Chrome 0.3+, Opera 9+, iOS1+, Safari 3.1+;
3. subsetting & compression comes down to author.



Web hosting & licensing services have sprung up offering a library of fonts at a variety of free and commercial plans (e.g. Typekit, Kernest, et al.).

## 8. Hosting & licensing services

Generally foundry initiatives that are js-based  
(though not exclusively) and hosted, paid  
subscription models: select, pay, inject; done.

## 8. Hosting & licensing services

Generally foundry initiatives that are js-based (though not exclusively) and hosted, paid subscription models: select, pay, inject; done.

Thoughts:

## 8. Hosting & licensing services

Generally foundry initiatives that are js-based (though not exclusively) and hosted, paid subscription models: select, pay, inject; done.

Thoughts:

1. deemed ‘the solution’ and becoming such;

## 8. Hosting & licensing services

Generally foundry initiatives that are js-based (though not exclusively) and hosted, paid subscription models: select, pay, inject; done.

Thoughts:

1. deemed ‘the solution’ and becoming such;
2. standards-based, good browser support, and easy;

## 8. Hosting & licensing services

Generally foundry initiatives that are js-based (though not exclusively) and hosted, paid subscription models: select, pay, inject; done.

Thoughts:

1. deemed ‘the solution’ and becoming such;
2. standards-based, good browser support, and easy;
3. subsetting and/or compression service-side.



# The easiest way to use Real Fonts ON YOUR WEBSITE

Bello by Underware

[Use this typeface ▶](#)

Sign up  
now!

## Tons of fonts, quick and easy setup

1 Pick a font



2 Paste javascript

```
1 <!DOCTYPE html PUBLIC  
2   "http://www.w3.org/1999/xhtml/DTD/xhtml1-strict.dtd">  
3 <html xmlns="http://  
4   www.w3.org/1999/xhtml">  
5 <head>  
6     <link rel="stylesheet" type="text/css" href="http://typekit.com/012345.css" />  
    <script type="text/javascript" src="http://typekit.com/012345.js" />  
  </head>
```

3 Voilà!



[Check out the fonts](#) or [Sign up](#)

Simple. Fast. Bulletproof.

Aa

Drop a single line of code in your web pages, and you can use whatever fonts you want—easy peasy. We host the files on a global network designed for uptime and speed so your fonts will always show up. It works with the latest browsers, and degrades gracefully for older ones. [Learn more](#)

Completely Legal

We work with some of the best type foundries in the world to make sure you and your clients can use fonts legally. [See the fonts](#)



Font FOUNDRY



Safe. Standards-compliant.



We use the W3C approved @font-face spec for CSS to link fonts to your site. We give you complete control over how the fonts interact with your markup. [Learn more](#)

[Launch Kit Editor](#)

## Skolar

[Sample](#)[Your text](#)

Have you ever wanted web design & development workshops to be a bit more, well, creative?

[EDIT](#)

### About this font

Although originally inspired by a wall lettering in a small chapel on Crete, Greece, it has nothing to do with it anymore. Despite its experimental character it works nicely in a text environment. Crete is perfect for display use where a feminine and elegant touch is desired. The unusual serifs and terminals add to the graceful appearance in the Thin and provide a more robust feel in the Thick. Both weights are metrically interchangeable, so text will not reflow when mixed. The accompanying Italic have several different lettershapes and therefore have, in some cases, their own widths. However, they sit comfortably next to the uprights. The names refer to the change in serif weight instead of increasing vertical stem widths. Crete was selected as winner of the Granshan competition 2008 in the display type category.

### About this foundry

Veronika Burian and José Scaglione met at the University of Reading whilst completing their MAs in Type Design, launching the independent type foundry TypeTogether in 2006.

The screenshot shows the Typekit Editor interface for the kit "For A Beautiful Web".

**Selectors (6)**

- .tk-skolar
- h1
- h2
- #nav-main a
- .aside .tel
- #introduction .one p

**Add**

**Welcome! Here's how to use fonts:**

In the left column, use the *Selectors* section to apply fonts to any **HTML tag, class, or id**. You can also add the default class to your markup. Then, click publish to see those changes on your site.

**Weights & Styles**  
Coming soon.

**Subsets**  
Coming soon.

**CSS Stack**  
Typekit will use the following font fallbacks for users whose browser doesn't support @font-face:  
Palatino, Georgia, Times, sans-serif  
**Save**

**Skolar by TypeTogether** [Delete this font](#)

AaBbCcDdEeFfGgHhIiJjKkLlMmNnOoP

**Kit Size:** 1 fonts, 6 selectors for For A Beautiful Web **Publish**

Our master-classes &  
upcoming dates

Read reviews of past  
workshops & events

Blogging about web design  
on And All That Malarkey

Get in touch with  
Stuff & Nonsense

# for a beautiful web

The brightest speakers, the most up-to-date web design & development information in a workshop atmosphere that makes learning creative. Say hello to For A Beautiful Web.

## Advanced CSS Styling

Join designer and author Andy Clarke for a full day learning how to apply the most up-to-date CSS in your work today. Be inspired to find new ways of using CSS.

Early-bird price £275 per person  
Standard price £325. Price excludes VAT.



### New UK dates announced

**BIRMINGHAM**  
**September 25th, 2009**

**REGISTER**

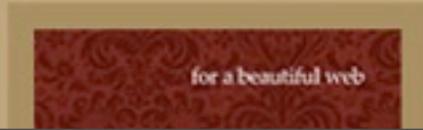
**NEWCASTLE UPON-TYNE**  
**October 30th, 2009**

**REGISTER**

Includes tuition, course materials, breakfast, lunch and snacks. Places are strictly limited.

New DVDs coming soon

Catch Andy Clarke on DVD in three new For



Home

Work

Blog

About

Projects

Contact

# Pixels, Creativity & Lorem Ipsum

Why not infuse your next project with a bit of  
German efficiency and Australian serenity?

Welcome to brizk design, the portfolio of German-born, Australian-based web design freelancer Kai Brach.

brizk design





Find fonts by

FONT  
DECK

Style

Designer

Foundry

# Rich web typography with Fontdeck

## Free to use and easy to set up

Set in [Centro Slab Pro](#) by [Parachute](#)

Fontdeck is the professional webfonts solution. You **only pay for the fonts you need, when you need them.** Get up and running in under 60 seconds.

[Browse fonts now](#)

### Browse our curated font catalogue →

Choose from 100s of professional typefaces from foundries including URW++ and Type Together.

### Add fonts to your site quickly and easily →

We'll deal out the fonts and produce the CSS; you simply copy and paste into your site.

### View all fonts in your website for free

Try any fonts you want, and share your site with up to 20 visitors: colleagues, clients and friends.

### Buy the perfect font deck

You've found the fonts that work for you. Purchase licenses on your final selections starting at only \$4.99 for one year.

## Featured Fonts

[Baskerville](#)

*Why pangolins dream of quich*

[Cooper Black](#)

**Why pangolins dream**

## Fast, Reliable & Scalable

- Use real fonts – no need for Flash, images or hacks
- Real text can be scaled, translated, read out loud and Googled
- Pure CSS @font-face solution – no JavaScript required
- Works on all modern browsers – IE, Firefox, Safari, Chrome, Opera, iPad iPhone
- No libraries or plans – get free access to all fonts from the start



# DOILESIC Passengers

Please **Charlie** is a typeface created for use in print and exhibition settings. An affable slab serif, Charlie's personality is clear and direct, yet warm and polite.

*n the upright, locked position*

SHA → AMS

Fly Madrid to Vancouver direct for only €157.00 plus tax

## Arrivals Arrivées

Charlie, a new slab serif typeface created for use in print and exhibition settings



### Font Collection

Brioni  
Brioni Sans  
Brioni Text  
Charlie  
Fedra Arabic  
Fedra Hindi  
Fedra Mono  
Fedra Mono Screen

Fedra Sans  
Fedra Sans Alt  
Fedra Sans Condensed  
Fedra Sans Condensed Alt  
Fedra Sans Display 1  
Fedra Sans Display 2  
Fedra Sans Screen  
Fedra Serif A

Fedra Serif B  
Fedra Serif Display  
Fedra Serif Phonetic  
Fedra Serif Screen  
Greta Display  
Greta Display Narrow  
Greta Grande  
Greta Grande Narrow

Greta Text  
Greta Text Narrow  
History  
Irma  
Irma Slab  
Jigsaw  
Jigsaw Stencil  
Klimax

Nara  
Plan Grot Cond Stencil  
Plan Grot Condensed  
Plan Grotesque  
Plan Grotesque Stencil

### Blog

#### Type specimens

MONDAY, 11 OCTOBER 2010

Peter Bil'ak, Eye magazine interview

MONDAY, 27 SEPTEMBER 2010

AGI-NL 2010

TUESDAY, 21 SEPTEMBER 2010

New typeface: Charlie

FRIDAY, 17 SEPTEMBER 2010

New typeface: Brioni Sans

FRIDAY, 17 SEPTEMBER 2010

Typotheque new website

FRIDAY, 17 SEPTEMBER 2010



## Price Calculator

Show prices in **EUR**

Family <b>Brioni Std</b>	Product <b>Medium Italic</b>	Users <b>1</b>	Licence <b>Web</b>	<b>18.00 EUR</b>	<b>Add</b>
-----------------------------	---------------------------------	-------------------	-----------------------	------------------	------------

**Calculation**

Item	# fonts	Options	Price	
<b>A Brioni Std Medium Italic</b>	<b>1</b>	<b>1 user — Web Licence</b>	<b>(18.00 × 1) 18.00 EUR</b>	<b>Remove</b>

**Subtotal (excl. VAT)** **18.00 EUR** **Place order**

[\*\*Save to account\*\*](#)
[\*\*Print quote\*\*](#)
[\*\*E-mail calculation\*\*](#)

- Std or Pro?**
- PostScript Type 1 fonts**
- Licensing**
- Ordering information**

Connect [Join our mailing list](#) [Submit](#) Follow us on twitter Become our fan on Facebook Check out our photos on Flickr Subscribe to RSS feed

Brioni	Fedra Mono Screen	Fedra Sans Screen	Greta Display Narrow	Irma Slab	Plan Grotesque
Brioni Sans	Fedra Sans	Fedra Serif A	Greta Grande	Jigsaw	Plan Grotesque Stencil
Brioni Text	Fedra Sans Alt	Fedra Serif B	Greta Grande Narrow	Jigsaw Stencil	
Charlie	Fedra Sans Condensed	Fedra Serif Display	Greta Text	Klimax	
Fedra Arabic	Fedra Sans Condensed Alt	Fedra Serif Phonetic	Greta Text Narrow	Nara	
Fedra Hindi	Fedra Sans Display 1	Fedra Serif Screen	History	Plan Grot Cond Stencil	
Fedra Mono	Fedra Sans Display 2	Greta Display	Irma	Plan Grot Condensed	

# Typotheque Web Font Service supports over 100 languages

## Font Subsetting

Regular Typotheque fonts support over 100 languages and advanced typographic features such as small caps, different numeral styles, alternative characters and special symbols. Web browsers, however, can access only a small fraction of those characters. Removing these inaccessible features makes the files much smaller, so they download much quicker. Users can specify which languages to support.

## Language support

Typotheque web fonts support all languages supported by their print versions.

## OpenType features

Typotheque monitors browser support of OpenType features will update its web fonts accordingly. Currently only Standard Ligatures and some special features required for non-Latin are supported in web fonts.

## Latin

Typotheque est une fonderie située à La Haye, Pays-Bas, qui développe et commercialise des polices de caractères

## Extended Latin

Typotheque jest domem typograficznym z siedzibą w Hadze w Holandii. Projektujemy oraz sprzedajemy fonty

## Greek

Η Typotheque είναι μία εταιρεία σχεδιασμού γραμμάτων με έδρα την Χάγη της Ολλανδίας, όπου

## Cyrillic

Типотек – шрифтовая фирма, находящаяся в Гааге (Нидерланды), разрабатывающая и реализующая

## Arabic

تيبوتيك شركة متخصصة في تصميم خطوط خاصة تصلح لتطبيقات ولغات متنوعة، إضافة إلى عملها على تكوين مكتبة للخطوط، في متناول

# Web Typography News<sup>1</sup>

## Main Headline<sup>2</sup>

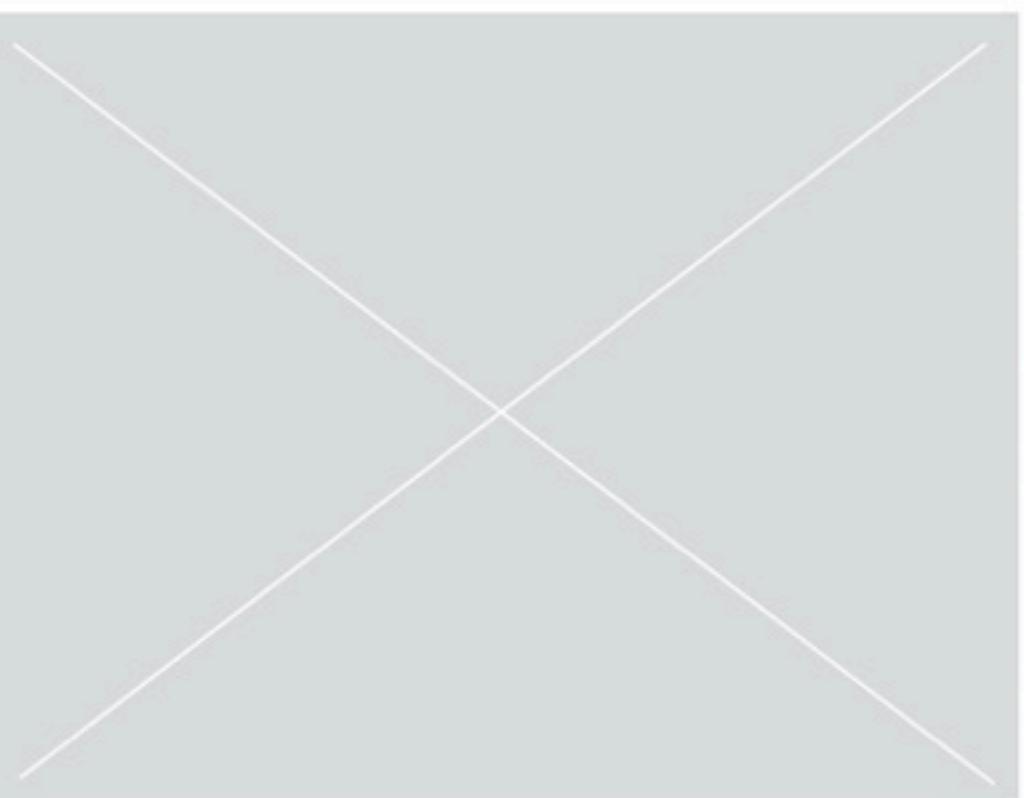
A smaller, secondary headline that usually elaborates on the main headline above it.<sup>3</sup>

<sup>4</sup> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Cras eu velit metus, imperdiet condimentum orci. Sed ac purus nulla. Vestibulum sed ligula ut nulla dignissim scelerisque. Nulla risus velit, feugiat quis sollicitudin non, ultricies id. Nunc vestibulum

Pellentesque bibendum eleifend imperdiet. Morbi elit tellus, volutpat sit amet malesuada egestas, vulputate et nisl. Sed nibh ipsum, varius non placerat a, tristique non odio. Suspendisse porta neque eu dui varius pretium hendrerit eget augue. Nullam pretium vestibulum nibh, quis molestie mi varius id. Donec ornare sem vitae erat congue interdum. Etiam id nisl eu erat pharetra convallis eget eu neque. Maecenas ut nisl in erat posuere iaculis. Aenean eget nisl in massa dignissim rhoncus quis eu lorem. In tempus, felis non bibendum semper, diam eros gravida.



fusce sed velit ipsum, sit amet malesuada orci. Nam mi lacus, pellentesque vitae condimentum non, sagittis interdum urna. Sed eu urna ut ipsum aliquam semper.



## Separate Article Title<sup>5</sup>

Mauris laoreet tincidunt enim, ac convallis odio laoreet nec. Suspendisse nec lectus arcu. Donec vestibulum pretium ipsum, at blandit mi scelerisque sed. In lobortis felis augue. In posuere aliquet velit et sem luctus et lobortis nibh varius. Vestibulum ante ipsum arcu, a vestibulum augue.

Etiam rhoncus velit quis turpis vehicula condimentum. Nulla eu volutpat dolor. Phasellus felis tellus, placerat gravida adipiscing non, posuere mattis augue. In posuere aliquet velit sed volutpat. Quisque sed risus arcu, a vestibulum augue.

Cras egestas gravida metus et egestas. Mauris eget ipsum sem. Nulla facilisi. Etiam luctus, mauris a ultrices semper, mi augue bibendum eros, metus nisi eu enim.

**"Short quote that summarizes the entire article".<sup>6</sup>**

Nulla vel sapien sit amet elit euismod imperdiet et nec turpis. Quisque feugiat sapien sit amet velit porta euismod. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed nec rhoncus risus. Phasellus sed magna nunc. Mauris tincidunt enim in elit volutpat dapibus. Donec sed ipsum eros. Donec pretium, tortor quis commodo dignissim, erat nisl fermentum orci, at convallis augue tellus eget neque. Praesent nec lorem consequat.

**FONTS**

- FONTCOLLECTION
- WEB FONTS
- TESTING FONTS
- COMBINING TYPE
- LANGUAGES
- FONTS IN USE
- IN DEVELOPMENT
- SERVICES

**LICENSING****BOOKS****T-SHIRTS****UTILITIES****ARTICLES****STUDIO WORK****ABOUT & CONTACT****MY ACCOUNT****HELP**

# BRINGING FINE TYPOGRAPHY TO THE WEB

The Typotheque Web Font Service enables you to use custom fonts in your website using the @font-face rule in CSS. Just add a line of code to your page and get it working in minutes. Simple, fast and standard-compliant.



## Frequently Asked Questions



- ✓ Works on 95% of browsers
- ✓ Multilingual support
- ✓ No JavaScript required
- ✓ Reliable cloud hosting
- ✓ Free Trial license
- ✓ One license for print & web
- ✓ One time license cost
- ✓ Advanced OpenType features
- ✓ W3C standard compliant

### Font Collection

- Brioni
- Brioni Text
- Fedra Arabic
- Fedra Mono
- Fedra Sans
- Fedra Sans Alt
- Fedra Sans Condensed
- Fedra Sans Condensed Alt
- Fedra Sans Display 1
- Fedra Sans Display 2
- Fedra Sans Hindi
- Fedra Serif A
- Fedra Serif B
- Fedra Serif Display
- Fedra Serif Phonetic
- Greta Display
- Greta Display Narrow
- Greta Grande
- Greta Grande Narrow
- Greta Text
- Greta Text Narrow
- History
- Irma
- Jigsaw
- Jigsaw Stencil
- Test

### Buying a license

To buy a license, go to the font page, click 'Buy this font' and choose the type of license you want.

## Pricing & Licensing

There are three types of licenses for Web fonts: the Full License gets you fonts for both print and web applications. The Web License is for web use only. The Trial License gives you 30 days free evaluation. See pricing ►

## Easy to implement

- 1 Choose a font license type
- 2 Create your Web Font
- 3 Copy & Paste the CSS code
- 4 Ready! Just refresh your page

## BUYING OPTIONS FOR FEDRA SANS

[OVERVIEW](#)   [ABOUT](#)   [CHARACTER SET](#)   [SAMPLE TEXT](#)   [PDF](#)   [FONT TESTER](#)   [FONTS IN USE](#)   [FREE BROCHURE](#)

Show prices in the following currency [EUR Euro](#)

License for 1 user(s)

[OpenType Fonts](#)

Typotheque OpenType fonts include Small Caps and different numeral styles as OpenType features, so they are not accounted as separate fonts.

**Fedra Sans Std (complete) contains 10 fonts:**

(Fedra Sans Std Light, Fedra Sans Std Light Italic, Fedra Sans Std Book, Fedra Sans Std Book Italic, Fedra Sans Std Normal, Fedra Sans Std Normal Italic, Fedra Sans Std Medium, Fedra Sans Std Medium Italic, Fedra Sans Std Bold, Fedra Sans Std Bold Italic)

**Fedra Sans Std (complete)**

10 fonts (13.80 per font)  
138.00 EUR

[Full License](#)  
[Web License](#) Web License [Trial Web License](#)

Extended Latin

**Fedra Sans Pro (complete)**

10 fonts (83.00 per font)  
830.00 EUR

[Full License](#)

Extended Latin, Greek, Cyrillic

**Fedra Sans Std Book Italic**

1 font  
90.00 EUR

[Full License](#)

Extended Latin

**Fedra Sans Std Medium**

1 font  
90.00 EUR

[Full License](#)

Extended Latin

**Fedra Sans Std Light**

1 font  
90.00 EUR

[Full License](#)

Extended Latin

Supported languages

- Abaza
- Adyghe
- Afrikaans
- Albanian
- Avar
- Azeri (Latin)
- Balkar
- Basque
- Belarusian
- Bosnian
- Breton
- Bulgarian
- Catalan
- Croatian
- Czech
- Danish
- Dargin
- Dutch
- English
- Esperanto
- Estonian
- Faroese
- Finnish
- French
- Friulian
- German
- Greek
- Greek (polytonic)
- Greenlandic
- Hawaiian
- Hungarian
- Icelandic
- Indonesian
- Ingush
- Interlingua
- Irish Gaelic
- Italian
- Kabardian
- Komi
- Kumyk
- Kurdish (Latin)
- Lak
- Latvian
- Lezgian
- Lithuanian
- Luxemburgish
- Macedonian
- Malay
- Maltese

## My Web Fonts

Here you can create embeddable fonts for use on the web. Simply fill in your specifications, and our system instantly generates a short block of CSS code. Paste the code into your website, and you are ready to start using the font. You will not work with actual font files, but with the CSS code linking to the font files.

Buy [more fonts here](#).

Project name

example: mysite.com

[EDIT](#)
[STATS](#)
[CODE](#)

[CREATE PROJECT](#)

There may be additional monthly costs for using Web Fonts. See [Pricing](#).

### Pricing

For every used web font you receive 500MB of free bandwidth per month. For a typical font file size (cca 20kB), that translates to 25,000 visitor views per month. Additional bandwidth costs €1 per extra Gigabyte.

### [Pricing Details](#)

You can monitor usage statistics, so you can know if you are exceeding the free monthly bandwidth.

Please read our [End User License](#).

### Status

- Currently in Use
- Fonts not Used
- Paused
- Budget Reached

Roll-over the icon to pause or to enable the project.

Please note that it takes 24 hours to correctly display the status of your project



FONTS  
LICENSING  
BOOKS  
T-SHIRTS  
UTILITIES  
ARTICLES

STUDIO WORK

ABOUT &amp; CONTACT

MY ACCOUNT

— MY ORDERS

— MY WEB FONTS

— PERSONAL INFO

— TYPE SPECIMENS

— LOG OUT

Hello Simon Klein

? HELP

## EDIT WEB FONT PROJECT

### 1 Please type in the name of your project:

example: mysite.com headline

### 2 Select the fonts

 Brioni Brioni Text Fedra Serif A Fedra Serif A Book

N B I BI

 Fedra Serif A Book Italic

N B I BI

 Fedra Serif A Demi

N B I BI

 Fedra Serif A Demi Italic

N B I BI

 Fedra Serif A Medium

N B I BI

 Fedra Serif A Medium Italic

N B I BI

 Fedra Serif A Bold

N B I BI

 Fedra Serif A Bold Italic

N B I BI

 Fedra Serif B Fedra Serif B Book

N B I BI

 Fedra Serif B Book Italic

N B I BI

 Fedra Serif B Demi

N B I BI

 Fedra Serif B Demi Italic

N B I BI

 Fedra Serif B Medium

N B I BI

 Fedra Serif B Medium Italic

N B I BI

 Fedra Serif B Bold

N B I BI

 Fedra Serif B Bold Italic

N B I BI

### 3 Define language(s) for font subsetting ?

Language: English [Add another language](#)German [Remove](#)

### 4 Select OpenType features ?

 Small Caps All Small Caps

Numerals: Old Style Numerals

### 5 Type in the domain where the fonts will be used ?

klepas.org

[Add another](#)

#### Making Web Fonts

1 Choose a name for your project. This name is for your internal purposes only and will be saved in your account.

2 Select the fonts which you have purchased and would like to use on the web.

3 Define the languages that you want to work with. The more languages you choose, the bigger the font file will be. To ensure fast downloading of the fonts all unnecessary glyphs will be removed from the fonts.

4 Select the OpenType features which will be applied to the font. This will generate a font using the OpenType feature you selected. You can't control these settings in your CSS, but you can edit these settings at any time.

5 Enter the URLs where you would like to use the fonts.



You're currently viewing a single article. Feel free to skip back to [Index](#) or straight to [Latest article](#).

[ABOUT & CONTACT](#)[ARCHIVES](#)[PORTFOLIO](#)[FEED](#) Search[← View previous article](#)[View next article →](#)

## *A critical examination of gender relations within goth subculture*

---

Published • 11 AUG 2009

---



INTER 2009, CHROME, CANBERRA, AUSTRALIA, A BLEND OF 'DARKWAVE' EBM (electronic body music) resounds from the club's speaker system. Black-clad dark outlines have formed on the dance floor beside the bar, gently fluctuating, writhing, and drifting to the industrial reverberations. The mysterious entourage flaunt a mixture of bulky leather boots, tight -fitting bondage wear, kilts, silky flowing evening gowns, capes, corsetry, and the odd fetish gear. The sheer volume of identities on display seems overwhelming: of the gender binaries both extremes of hyper-masculinities and -femininities are represented with a vast variation of graded representations between them. However apparent also is small curious overlap that seems neither distinctly masculine nor feminine. What could appear to outsiders as an odd, clandestine group of adolescent misfits is the monthly popular goth event of Canberra city, a type of gathering that occurs in commonality weekly across the world.

# Using @font-face

# Using @font-face

## 0. Reset

# Using @font-face

## 0. Reset

All the browsers have their own default styling for various (x)HTML elements. This makes it a pain for cross-browser consistency.

```
body,div,dl,dt,dd,ul,ol,li,h1,h2,h3,h4,h5,h6,pre,form,fieldset,input,textarea,p,blockquote,th,td {margin:0;padding:0;}  
table {border-collapse:collapse; border-spacing:0;}  
fieldset,img {border:0;}  
address,caption,cite,code,dfn,em,strong,th,var {font-style: normal;font-weight: normal;}  
ul,ol {list-style:none;}  
caption,th {text-align:left;}  
h1,h2,h3,h4,h5,h6 {font-size:100%;font-weight: normal;}  
q:before,q:after {content:'';}  
abbr,acronym {border:0;}
```

Googling “css reset stylesheets” should land you with numerous examples. This one above is from the Yahoo! UI library.

## ‘bullet-proof’ @font-face declarations

## ‘bullet-proof’ @font-face declarations

```
@font-face {
```

## ‘bullet-proof’ @font-face declarations

```
@font-face {  
    font-family: "Your typeface";
```

## ‘bullet-proof’ @font-face declarations

```
@font-face {  
    font-family: "Your typeface";  
    src: url("type/filename.eot");
```

## ‘bullet-proof’ @font-face declarations

```
@font-face {  
    font-family: "Your typeface";  
    src: url("type/filename.eot");  
    src: local("Alt name"), local("Alt-name"),
```

## ‘bullet-proof’ @font-face declarations

```
@font-face {  
    font-family: "Your typeface";  
    src: url("type/filename.eot");  
    src: local("Alt name"), local("Alt-name"),  
        url("type/filename.woff") format("woff"),
```

## ‘bullet-proof’ @font-face declarations

```
@font-face {  
    font-family: "Your typeface";  
    src: url("type/filename.eot");  
    src: local("Alt name"), local("Alt-name"),  
        url("type/filename.woff") format("woff"),  
        url("type/filename.otf") format("opentype"),
```

## ‘bullet-proof’ @font-face declarations

```
@font-face {  
    font-family: "Your typeface";  
    src: url("type/filename.eot");  
    src: local("Alt name"), local("Alt-name"),  
        url("type/filename.woff") format("woff"),  
        url("type/filename.otf") format("opentype"),  
        url("type/filename.svg#id") format("svg");  
}
```

When using `@font-face`, we're essentially dealing with separate font files of the same family for the various font styles, e.g., `foobar-regular.otf`, `foobar-italic.otf`, `foobar-bold.otf`, `foobar-smallcaps.otf` &c.

When using `@font-face`, we're essentially dealing with separate font files of the same family for the various font styles, e.g., `foobar-regular.otf`, `foobar-italic.otf`, `foobar-bold.otf`, `foobar-smallcaps.otf` &c.

This can become an issue—consider elements such as `strong` and `em` which are styled with setting the face in a bold and italic respectively.

When using `@font-face`, we're essentially dealing with separate font files of the same family for the various font styles, e.g., `foobar-regular.otf`, `foobar-italic.otf`, `foobar-bold.otf`, `foobar-smallcaps.otf` &c.

This can become an issue—consider elements such as `strong` and `em` which are styled with setting the face in a bold and italic respectively.

If we declare the italic via `@font-face` (as we would to get the style into our design) what will happen is that *the italic is digitally italicised* (fake italics) by the font rendering engine. Result?

- Grunerberg, C. 1997, *Unsolved Mysteries: Gothic Tales from Frankenstein to the Hair Eating Doll*, MIT Press, Boston.
- Holmes, J. 2003, *Crimson out of the Coffin: Gay Males and Queer Goths in Contemporary Vampire Fiction*, University of Pennsylvania Press, Philadelphia.
- Leblanc, N. 2002, *Sexuality and Gender*, pp. 167–173) Blackwell, Oxford (UK).
- Macdonald, N. 2001, *The Graffiti subculture. Youth masculinity, and femininity in London and New York*, Palgrave, Basingstoke.
- Skeggs, B. 1997, *Formations of Class and Gender: Becoming Respectable*, SAGE, London.
- Wilkins, A. C. 2004, "So Full of Myself as a Chick": Goth Women, Sexual Independence, and Gender Egalitarianism, *Gender & Society*, SAGE Publications, Inc., Vol. 18, No. 3, 331–342.

Download a PDF, printer-friendly version of this article.

For referencing: [permalink to this article](#).

[← View previous article](#)

[View next article →](#)

---

[ABOUT & CONTACT](#)

[NOTEBOOK](#)

[PORTFOLIO](#)

[FEED](#)

[Back to top ↑](#)

 Add me to your address book via Card.

Proudly bending beziers since MMIV—; copyright © Simon Pascal Klein thereon.

Some rights reserved.

For an overview of the site, see the [site map](#) or [notebook archives](#).

A playful tumble with **type**... —

yes, **type**, not **graphic**; try selecting and resizing the page font size. Inspired by Jon Tan's masterhead type folly, this is achieved just with some simple nested **span** elements that are positioned absolutely against their relative parents, and the entire thing is sized in ems and scales nicely.



...all in mostly valid CSS & XHTML; as posh as can be.

If we avoid or overwrite the various declarations (e.g. `em { font-style: normal; }`) and for whatever reason our desired `@font-face` font isn't available, we rob other fonts in the font-stack of their styling.

We overcome both these issues by setting the font styles within the `@font-face` declaration:

```
@font-face {  
    font-family: "Your italic typeface";  
    src: url("type/filename.eot");  
    src: local("Alt italic name"), local("Alt-italic-name"),  
        url("type/filename-italic.woff") format("woff"),  
        url("type/filename-italic.otf") format("opentype"),  
        url("type/filename-italic.svg#id") format("svg");  
}
```

We overcome both these issues by setting the font styles within the `@font-face` declaration:

```
@font-face {  
    font-family: "Your italic typeface";  
  
    src: url("type/filename.eot");  
    src: local("Alt italic name"), local("Alt-italic-name"),  
        url("type/filename-italic.woff") format("woff"),  
        url("type/filename-italic.otf") format("opentype"),  
        url("type/filename-italic.svg#id") format("svg");  
}
```

We overcome both these issues by setting the font styles within the `@font-face` declaration:

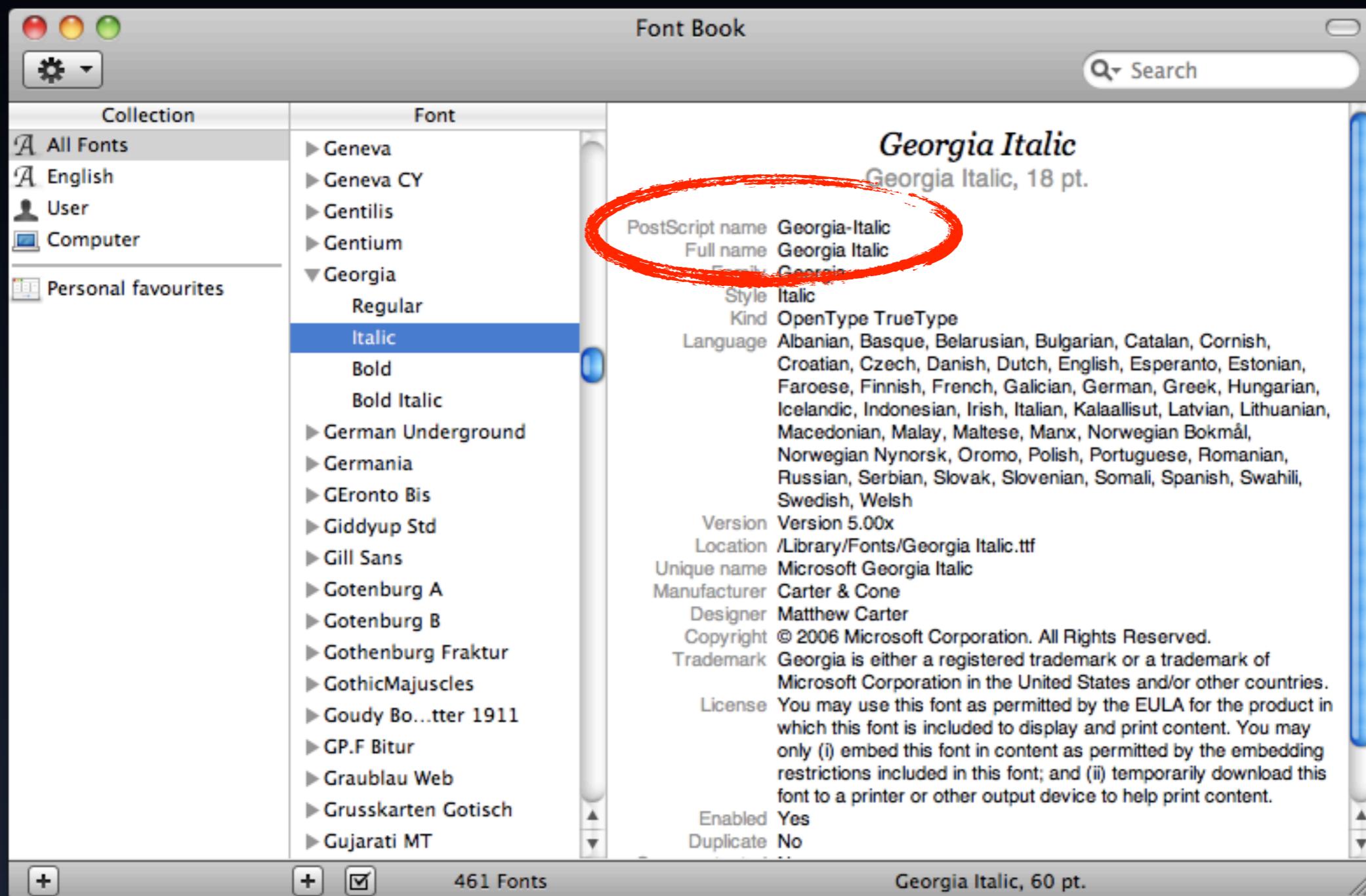
```
@font-face {  
    font-family: "Your italic typeface";  
    font-style: italic;  
    src: url("type/filename.eot");  
    src: local("Alt italic name"), local("Alt-italic-name"),  
        url("type/filename-italic.woff") format("woff"),  
        url("type/filename-italic.otf") format("opentype"),  
        url("type/filename-italic.svg#id") format("svg");  
}
```

Why are there two entries for local in src?

src: local("Alt italic name"), local("Alt-italic-name"),

Some browsers refer to the fonts using their PostScript names.

## Finding it in Font Book under Mac OS X:



There is a free downloadable **Font properties** extension for Windows. Once installed *right-click* → *Properties* on a font file and click on the *Names* tab to see the name details.

There is a free downloadable **Font properties** extension for Windows. Once installed *right-click* → *Properties* on a font file and click on the *Names* tab to see the name details.

See: <http://bit.ly/ms-font-prop>.

There is a free downloadable **Font properties** extension for Windows. Once installed *right-click* → *Properties* on a font file and click on the *Names* tab to see the name details.

See: <http://bit.ly/ms-font-prop>.

There are similar tools on the free desktops.



When using `@font-face` for *self-hosted* fonts, we  
need to **subset** and **compress** ourselves.

When using `@font-face` for *self-hosted* fonts, we need to **subset** and **compress** ourselves.

Subsetting allows us to remove knowingly unused glyphs from the font file, thus cutting down file size. E.g. if we know we won't ever need Greek characters we can remove them reducing the font file's size, thus lowering page size.

When using `@font-face` for *self-hosted* fonts, we need to **subset** and **compress** ourselves.

Subsetting allows us to remove knowingly unused glyphs from the font file, thus cutting down file size. E.g. if we know we won't ever need Greek characters we can remove them reducing the font file's size, thus lowering page size.

Subsetting is can be done in a font editor—I suggest **FontForge** (FOSS)—open your font, select unused character blocks and delete them; save.



Save yourself a headache—skip using the `WEFT` converter and instead use `ttf2eot`. Why?

Save yourself a headache—skip using the WEFT converter and instead use **ttf2eot**. Why?

1. WEFT will use Monotype's proprietary compression algorithm and gzip is better;

Save yourself a headache—skip using the WEFT converter and instead use [ttf2eot](#). Why?

1. WEFT will use Monotype's proprietary compression algorithm and gzip is better;
2. WEFT only works on Windows XP(?);

Save yourself a headache—skip using the WEFT converter and instead use **ttf2eot**. Why?

1. WEFT will use Monotype's proprietary compression algorithm and gzip is better;
2. WEFT only works on Windows XP(?);
3. ttf2eot doesn't compress—use gzip server-side.

Save yourself a headache—skip using the WEFT converter and instead use **ttf2eot**. Why?

1. WEFT will use Monotype's proprietary compression algorithm and gzip is better;
2. WEFT only works on Windows XP(?);
3. ttf2eot doesn't compress—use gzip server-side.

There are online front ends to **ttf2eot/sfnt2woff**:

Save yourself a headache—skip using the WEFT converter and instead use **ttf2eot**. Why?

1. WEFT will use Monotype's proprietary compression algorithm and gzip is better;
2. WEFT only works on Windows XP(?);
3. ttf2eot doesn't compress—use gzip server-side.

There are online front ends to **ttf2eot/sfnt2woff**:

- <http://ttf2eot.sebastiankippe.com/>

Save yourself a headache—skip using the WEFT converter and instead use **ttf2eot**. Why?

1. WEFT will use Monotype's proprietary compression algorithm and gzip is better;
2. WEFT only works on Windows XP(?);
3. ttf2eot doesn't compress—use gzip server-side.

There are online front ends to **ttf2eot/sfnt2woff**:

- <http://ttf2eot.sebastiankippe.com/>
- <http://fontsquirrel.com/fontface/generator>

# @font-face Kit Generator

Add Fonts

FifthLeg Regular

OTF

257 glyphs

87 KB

Aspect Ratio: 0.48



EASY

Target all browsers with ideal settings combination.

EXPERT...

Take complete control of the settings.

Font Formats:

TrueType

EOT

SVG

WOFF

SVGZ

Font Options:

Add Hinting

Improve Win rendering

WebOnly™

Disable desktop use

Keep OT Features\*

Ligatures, alt glyphs etc.

Remove Kerning

Strip kerning Data

Simplify Outlines

Remove extra points (lossy)

Build Cufón File

Javascript font alternative

Subsetting:

Basic Subsetting

Western languages

Custom Subsetting...

Custom language support

No Subsetting

CSS Formats:

Bulletproof (Smiley)

[Modified Bulletproof](#)

Avoids local() problems

Bulletproof (Original)

[Paul Irish Method](#)

Safe and compact

Mo' Bulletproofer

[Richard Fink Method](#)

Uses double declarations

CSS Options:

Style Linking

Group styles under family

Base64 Encode

Encode Font Within CSS

Agreement:

Yes, the fonts I'm uploading are legally eligible for web embedding.

Font Squirrel offers this service in good faith. Please honor the EULAs of your fonts.

For compression, let's gzip (via .htaccess):

For compression, let's gzip (via .htaccess):

```
# gzips content if possible
<IfModule mod_gzip.c>
    mod_gzip_on           Yes
    mod_gzip_dechunk     Yes
    mod_gzip_item_include file  \.(html?|css|js|...|otf|eot)$
    mod_gzip_item_include handler ^cgi-script$
    ...
    mod_gzip_item_include mime   ^application/vnd.ms-fontobject$
```

</IfModule>

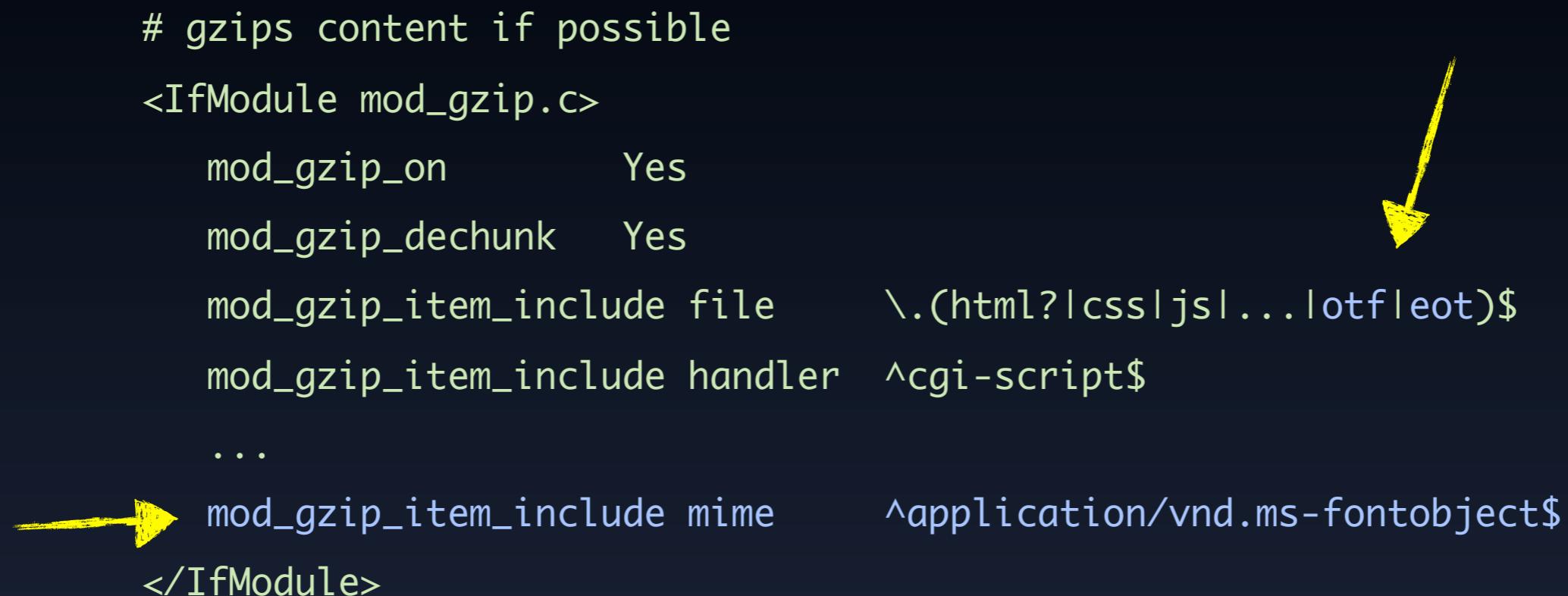
For compression, let's gzip (via .htaccess):

```
# gzips content if possible
<IfModule mod_gzip.c>
    mod_gzip_on           Yes
    mod_gzip_dechunk     Yes
    mod_gzip_item_include file  \.(html?|css|js|...|otf|eot)$
    mod_gzip_item_include handler ^cgi-script$
    ...
    mod_gzip_item_include mime   ^application/vnd.ms-fontobject$
```



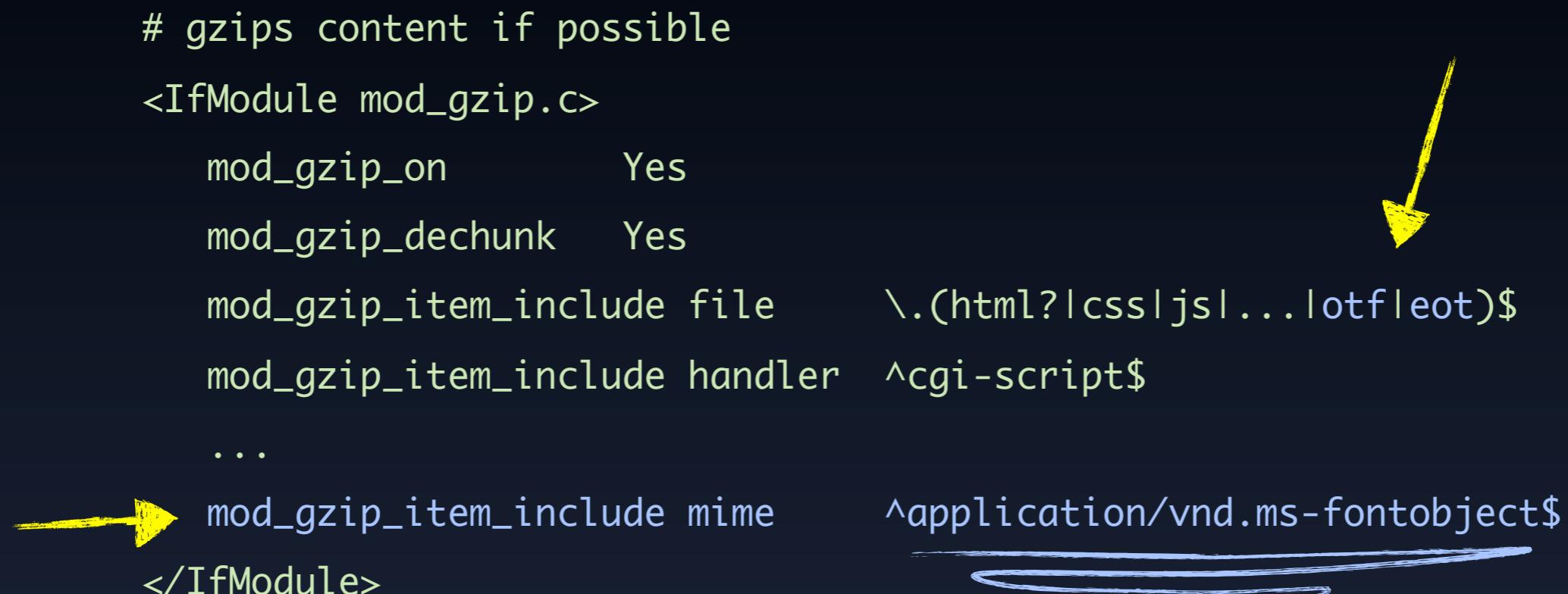
For compression, let's gzip (via .htaccess):

```
# gzips content if possible
<IfModule mod_gzip.c>
    mod_gzip_on           Yes
    mod_gzip_dechunk     Yes
    mod_gzip_item_include file  \.(html?|css|js|...|otf|eot)$
    mod_gzip_item_include handler ^cgi-script$
    ...
    → mod_gzip_item_include mime    ^application/vnd.ms-fontobject$
</IfModule>
```



For compression, let's gzip (via .htaccess):

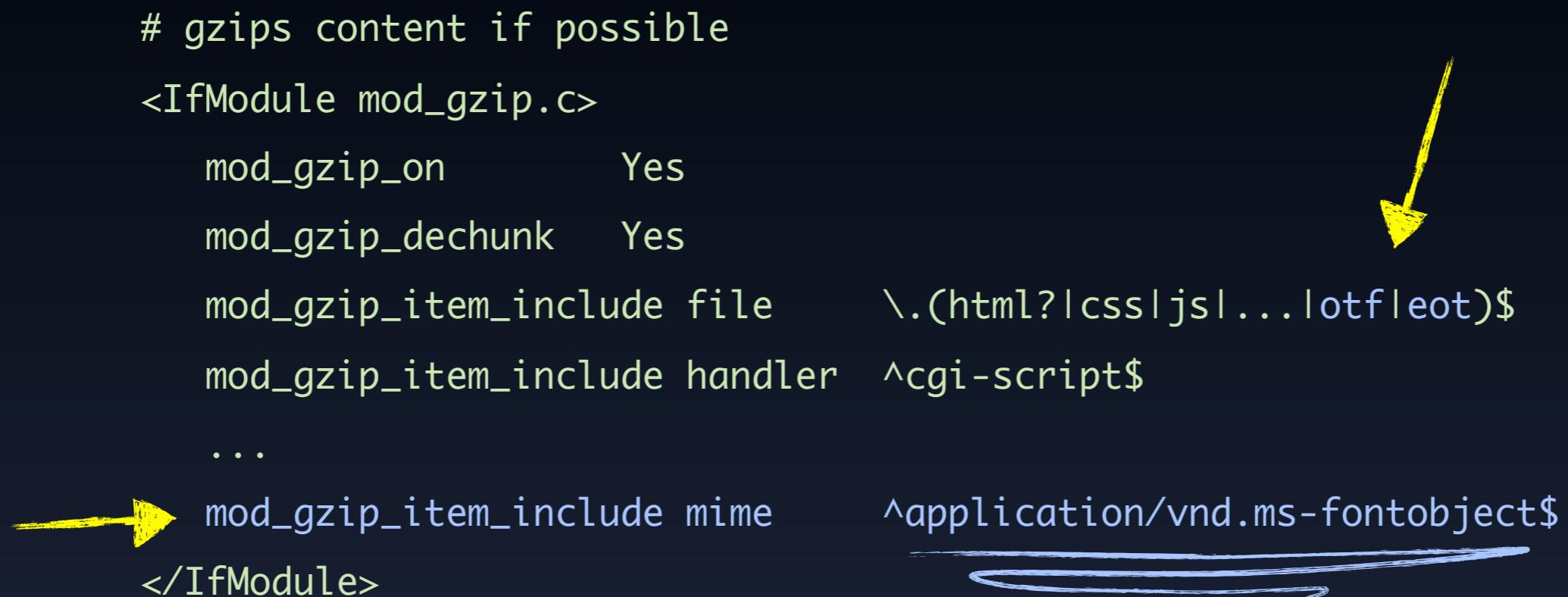
```
# gzips content if possible
<IfModule mod_gzip.c>
    mod_gzip_on           Yes
    mod_gzip_dechunk     Yes
    mod_gzip_item_include file  \.(html?|css|js|...|otf|eot)$
    mod_gzip_item_include handler ^cgi-script$
    ...
    mod_gzip_item_include mime   ^application/vnd.ms-fontobject$
```



A yellow arrow points from the left margin to the 'mod\_gzip\_item\_include mime' line. A blue wavy line underlines the '^application/vnd.ms-fontobject\$' part of that line.

For compression, let's gzip (via .htaccess):

```
# gzips content if possible
<IfModule mod_gzip.c>
    mod_gzip_on           Yes
    mod_gzip_dechunk     Yes
    mod_gzip_item_include file  \.(html?|css|js|...|otf|eot)$
    mod_gzip_item_include handler ^cgi-script$
    ...
    mod_gzip_item_include mime   ^application/vnd.ms-fontobject$
```



Don't be surprised to see reductions of 40-50%.

While we're at it, let's cache the font assets too:

While we're at it, let's cache the font assets too:

```
# Cache following file types for one month
<FilesMatch ".(js|jpeg|jpg|...|otf|eot)$">
    Header set Cache-Control "max-age=2592000"
</FilesMatch>
```

While we're at it, let's cache the font assets too:

```
# Cache following file types for one month
<FilesMatch ".(js|jpeg|jpg|...|otf|eot)$"> ←
    Header set Cache-Control "max-age=2592000"
</FilesMatch>
```

Finding fonts...?

# Finding fonts...?

- <http://openfontlibrary.org/>

# Finding fonts...?

- <http://openfontlibrary.org/>
- <http://www.theleagueofmoveabletype.com/>

# Finding fonts...?

- <http://openfontlibrary.org/>
- <http://www.theleagueofmoveabletype.com/>
- <http://www.fontsquirrel.com/fontface>

# Finding fonts...?

- <http://openfontlibrary.org/>
- <http://www.theleagueofmoveabletype.com/>
- <http://www.fontsquirrel.com/fontface>
- list on [webfonts.info](http://webfonts.info): <http://bit.ly/16Qid4>

2



# Meanwhile:

# Meanwhile:

Webfonts aren't the holy grail of web  
typography problems...

## Meanwhile:

Webfonts aren't the holy grail of web  
typography problems...

“Typography is *not* picking a ‘cool’ font.”

- Jeff Croft

Also remember, most fonts available from type foundries are not optimised for screen usage; they're for print.



So... now having selected our desired typeface (via webfonts or otherwise), let's make it pretty.



serif

serif

sans serif

serif

sans serif

*script*

serif

sans serif

*script*

blackletter

serif

sans serif

*script*

blackletter

monospace



Arial

Verdana

Courier New

Times New Roman

Georgia

Trebuchet MS



Use a limited palette of type families. A common and effective technique is to pair a serif and a sans serif face together. E.g.:

Use a limited palette of type families. A common and effective technique is to pair a serif and a sans serif face together. E.g.:

1. one sets the body, the other the headings;

Use a limited palette of type families. A common and effective technique is to pair a serif and a sans serif face together. E.g.:

1. one sets the body, the other the headings;
2. one sets primary content, the other UI controls.

Use a limited palette of type families. A common and effective technique is to pair a serif and a sans serif face together. E.g.:

1. one sets the body, the other the headings;
2. one sets primary content, the other UI controls.

We do this with the `font-family` property.



Use a combination of families, styles, and weights  
to breath some fresh air into your web type.

Use a combination of families, styles, and weights  
to breath some fresh air into your web type.

- `font-weight` controls weight and accepts numerical and descriptive (e.g. `normal`) values.

Use a combination of families, styles, and weights to breath some fresh air into your web type.

- `font-weight` controls weight and accepts numerical and descriptive (e.g. `normal`) values.
- `font-style` controls... style. Accepts the descriptive values (`normal`, `italic`, and `oblique`).

Use a combination of families, styles, and weights to breath some fresh air into your web type.

- `font-weight` controls weight and accepts numerical and descriptive (e.g. `normal`) values.
- `font-style` controls... style. Accepts the descriptive values (`normal`, `italic`, and `oblique`).
- `font-variant` controls case and accepts the values `normal` and `small-caps`.



roman ('normal')

roman ('normal')

roman ('normal')

*italic*

roman ('normal')

*italic* ≠ *oblique*

ultralight regular bold

roman ('normal')

*italic* ≠ *oblique*

ultralight regular bold

roman ('normal')

*italic* ≠ *oblique*

ultralight regular bold

& *bold italic condensed*

roman ('normal')

*italic* ≠ *oblique*

ultralight regular bold

& *bold italic condensed*

roman ('normal')

*italic* ≠ *oblique*

ultralight regular bold

& *bold italic condensed*

SMALL-CAPITALS

roman ('normal')

*italic* ≠ *oblique*

ultralight regular bold

& *bold italic condensed*

SMALL-CAPITALS



# TRACKING

TRACKING  
TRACKING

TRACKING  
TRACKING  
Letter-spacing

TRACKING

TRACKING

Letter-spacing

Letter-spacing



Let's see some loose word-spacing in action...

Let's see some loose word-spacing in action...

And now both loose word-spacing and tracking featured together.



**And finally, the recently popular  
trend of negative tracking.**



Tracking and word-spacing can help differentiate,  
aiding readability by removing disruption.

Tracking and word-spacing can help differentiate,  
aiding readability by removing disruption.

E.g. setting acronyms or numerical tabular data.



Full-capital acronyms, along with other full-cap-sized glyphs can disturb the flow of the text:

Full-capital acronyms, along with other full-cap-sized glyphs can disturb the flow of the text:

  Lorem ipsum dolor sit amet, NASA consectetur adipiscing et. Vestibulum elit pellentesque; ac habitant orci tristique senectus et 4,962 malesuada fames egestas HTML sit lectus.

Full-capital acronyms, along with other full-cap-sized glyphs can disturb the flow of the text:

  Lorem ipsum dolor sit amet, NASA consectetur adipiscing et. Vestibulum elit pellentesque; ac habitant orci tristique senectus et 4,962 malesuada fames egestas HTML sit lectus.

Full-capital acronyms, along with other full-cap-sized glyphs can disturb the flow of the text:

  Lorem ipsum dolor sit amet, NASA consectetur adipiscing et. Vestibulum elit pellentesque; ac habitant orci tristique senectus et 4,962 malesuada fames egestas HTML sit lectus.

  Lorem ipsum dolor sit amet, NASA consectetur adipiscing et. Vestibulum elit pellentesque; ac habitant orci tristique senectus et 4,962 malesuada fames egestas HTML sit lectus.

Full-capital acronyms, along with other full-cap-sized glyphs can disturb the flow of the text:

Lorem ipsum dolor sit amet, NASA consectetur adipiscing et. Vestibulum elit pellentesque; ac habitant orci tristique senectus et 4,962 malesuada fames egestas HTML sit lectus.

Lorem ipsum dolor sit amet, NASA consectetur adipiscing et. Vestibulum elit pellentesque; ac habitant orci tristique senectus et 4,962 malesuada fames egestas HTML sit lectus.



- `letter-spacing` and `word-spacing` controls tracking and word-spacing respectively. Both take numerical values and `normal`.

- `letter-spacing` and `word-spacing` controls tracking and word-spacing respectively. Both take numerical values and `normal`.

Sadly no property for controlling figure variants—hopes set for CSS3. For now use Georgia which (only) features hanging or ‘old-style’ figures.



- **font-size** controls... font sizes. Takes numerical and descriptive values. Most user agents (e.g. browsers) set default value at 16px.

- **font-size** controls... font sizes. Takes numerical and descriptive values. Most user agents (e.g. browsers) set default value at 16px.

Size font sizes relatively using ems or %.

- **font-size** controls... font sizes. Takes numerical and descriptive values. Most user agents (e.g. browsers) set default value at 16px.

Size font sizes relatively using ems or %.

An em is a relative unit best to be thought of as a box. It is relative to the point size of a specific font (i.e. 1 em in a 12 point typeface is 12 point).

# Why?

## Why?

- Some UAs don't support font-resizing (Ctrl + +/-) or page scaling when measurements are defined in absolute terms.

## Why?

- Some UAs don't support font-resizing (Ctrl + +/-) or page scaling when measurements are defined in absolute terms.
- JS-based text resize widgets ≠ text accessibility.

## Why?

- Some UAs don't support font-resizing (Ctrl + +/-) or page scaling when measurements are defined in absolute terms.
- JS-based text resize widgets ≠ text accessibility.
- Many devices and many user agents—be careful in making assumptions.



**Key:** remember font sizes are inherited:  
parents → children.

**Key:** remember font sizes are inherited:  
parents → children.

To calculate, find what one pixel is in ems and  
then multiply by the desired font size (in pixels):

Key: remember font sizes are inherited:  
parents → children.

To calculate, find what one pixel is in ems and  
then multiply by the desired font size (in pixels):

$$\begin{aligned} 1 \div \text{parent font-size} \times \text{required pixel value} \\ = \text{em value} \end{aligned}$$

62.5% trick

## 62.5% trick

Simplify your calculations. Consider:

```
p { font-size: 80%; }  
blockquote { font-size: 80%; }
```

## 62.5% trick

Simplify your calculations. Consider:

```
p { font-size: 80%; }  
blockquote { font-size: 80%; }
```

80% of 16px is 12.8px, so `p` and `blockquote` elements will be that size, but what happens when we put a `p` element inside a `blockquote` element? The parent (`blockquote`) is 12.8px so the `p` will be rendered at 80% of that: 10.42px.

## 62.5% trick (cont'd)

In 2004 Richard Rutter wrote an article outlining a trick he used to make these calculations simpler:

## 62.5% trick (cont'd)

In 2004 Richard Rutter wrote an article outlining a trick he used to make these calculations simpler:

- browsers have common default size of 16px; thus:

## 62.5% trick (cont'd)

In 2004 Richard Rutter wrote an article outlining a trick he used to make these calculations simpler:

- browsers have common default size of 16px; thus:
- set body to 62.5% and reset all to 10px; thus:

## 62.5% trick (cont'd)

In 2004 Richard Rutter wrote an article outlining a trick he used to make these calculations simpler:

- browsers have common default size of 16px; thus:
- set body to 62.5% and reset all to 10px; thus:
- from here calculations are simpler for direct descendants of the body, e.g. 12px = 1.2em; 8px = 0.8em. Further nested elements are (still) relative.



To find an element's font size in absolute terms (e.g. pixels) you can use the Firefox 'Web Developer' plugin: *Information* → *Display Element Information*.

Mixing typefaces requires special attention. Try to align x-heights and sizes. X-heights?



Helvetica Regular 96pt & Dolly Regular 96pt.

Mixing typefaces requires special attention. Try to align x-heights and sizes. X-heights?



x-height x-height

Helvetica Regular 96pt & Dolly Regular 96pt.



x-height x-height

Helvetica Regular 96pt & Dolly Regular 108pt.

# Stick it to a scale

Don't just arbitrarily set type; use a scale:

## Stick it to a scale

Don't just arbitrarily set type; use a scale:

6, 7, 8, 9, 10, 11, 12, 14, 16, 18, 21, 24, 36, 48, 60, 72.

*the “classic scale”*

9, 10, 11, 12, 13, 14, 18, 24, 36, 48, 64, 72, 96.

*another scale*

8, 13, 21, 34, 55, 89.

*scale based on the Fibonacci sequence*



At small sizes `font-decoration: underline;`  
can render descender glyphs (g, j, p, q, y) difficult  
to read: use `border-bottom: 1px solid;`

At small sizes `font-decoration: underline;`  
can render descender glyphs (g, j, p, q, y) difficult  
to read: use `border-bottom: 1px solid;`

...consectetuer adipiscing elit...

...adipiscing elit...

At small sizes `font-decoration: underline;`  
can render descender glyphs (g, j, p, q, y) difficult  
to read: use `border-bottom: 1px solid;`

...consectetuer adipiscing elit...

...adipiscing elit...

...consectetuer adipiscing elit...

...adipiscing elit...

# Correct glyphs

Consider:

! " "

# Correct glyphs

Consider:

1'61"

"This is a group of words  
surrounded by a bunch  
of tick marks."

“This is a quotation!”

- Jeff Croft

“ ” “ ”  
‘ ’



- hyphen      Mary-Anne, Walter-Strauss
- en dash      April-May, ages 3-5, pp. 37-38
- em dash      Phrase marker—(generally) not spaced—like so.
- minus       $5-4=1$

“	opening double quote	&ldquo;
”	closing double quote	&rdquo;
‘	opening single quote	&lsquo;
’	closing single quote	&rsquo;
-	en dash	&ndash;
—	em dash	&mdash;
–	minus	&minus;
×	multiplication	&times;
...	ellipsis	&hellip;

# *Smartypants et al.*

# *Smartypants et al.*

*'Doing something repetitively? Likely you're doing it wrong.'*

## *Smartypants et al.*

*‘Doing something repetitively? Likely you’re doing it wrong.’*

Smartypants et al. are parsers that translate plain ASCII characters into “smart” typographic (x)HTML entities (and beyond just punctuation).

## *Smarty pants et al.*

*‘Doing something repetitively? Likely you’re doing it wrong.’*

Smarty pants et al. are parsers that translate plain ASCII characters into “smart” typographic (x)HTML entities (and beyond just punctuation).

- Smarty pants (PHP, Perl, & Movable Type);

## *Smarty pants et al.*

*‘Doing something repetitively? Likely you’re doing it wrong.’*

Smarty pants et al. are parsers that translate plain ASCII characters into “smart” typographic (x)HTML entities (and beyond just punctuation).

- Smarty pants (PHP, Perl, & Movable Type);
- Typogrify (Python/Django);

## *Smarty pants et al.*

*‘Doing something repetitively? Likely you’re doing it wrong.’*

Smarty pants et al. are parsers that translate plain ASCII characters into “smart” typographic (x)HTML entities (and beyond just punctuation).

- Smarty pants (PHP, Perl, & Movable Type);
- Typogrify (Python/Django);
- wp-Typography (WordPress);

# *Smarty pants et al.*

*‘Doing something repetitively? Likely you’re doing it wrong.’*

Smarty pants et al. are parsers that translate plain ASCII characters into “smart” typographic (x)HTML entities (and beyond just punctuation).

- Smarty pants (PHP, Perl, & Movable Type);
- Typogrify (Python/Django);
- wp-Typography (WordPress);
- Markdown, Textile et al., ...



Give ampersands love

# Give ampersands love

Italic ampersand variants are often much prettier  
than their common roman counterparts:

# Give ampersands love

Italic ampersand variants are often much prettier  
than their common roman counterparts:

& & & & & &

# Give ampersands love

Italic ampersand variants are often much prettier  
than their common roman counterparts:

& & & & & &

& & & & & &

# Marking paragraphs (and more)

# Marking paragraphs (and more)

Don't be afraid to mark new paragraphs with  
indents, outdents, white-lines, a pilcrow (¶) or  
other ornament (e.g. ♥), versals, headers &c.

# Marking paragraphs (and more)

Don't be afraid to mark new paragraphs with  
indents, outdents, white-lines, a pilcrow (¶) or  
other ornament (e.g. ☙), versals, headers &c.

...or a combination thereof.

## Marking paragraphs (and more)

Don't be afraid to mark new paragraphs with  
indents, outdents, white-lines, a pilcrow (¶) or  
other ornament (e.g. ☙), versals, headers &c.

...or a combination thereof.

See: <http://tinyurl.com/para-typography>.

# Measures

The measure is the length of a single line. It is important to select a good measure for running text. Do this with the width property.

## Measures

← →  
The measure is the length of a single line. It is important to select a good measure for running text. Do this with the `width` property.

← →  
Ideally these should be relative to the font size, such that the type scales proportionately to the measure; use ems or percentages.

# Measures (cont'd)

## Measures (cont'd)

Web type generally benefits from smaller measures than those in print. A good ballpark is **10-15 words/75-100 characters** per line for body copy, single column.

## Measures (cont'd)

Web type generally benefits from smaller measures than those in print. A good ballpark is **10-15 words/75-100 characters** per line for body copy, single column.

Generally use flush-left (`text-align: left;`) for running text, particularly when set in a narrow measure. Justification can work at ample measures and better with serif typefaces.

# Leading (line-height)

# Leading (line-height)

Don't forget to set an ample leading for running text! Done using the `line-height` property and you can use a unit-less number (e.g. 1.5 which acts as a multiplier of the font size);

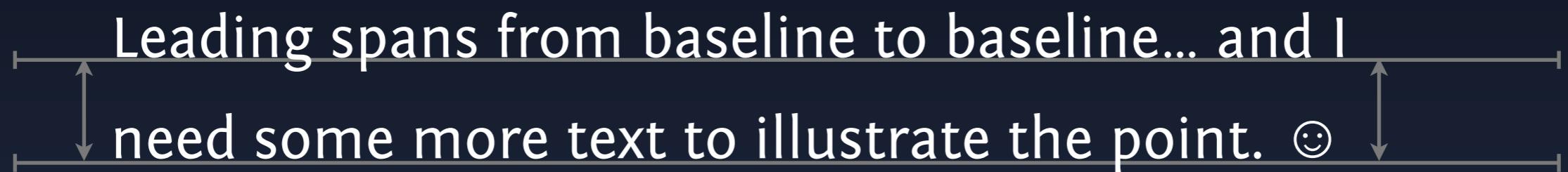
## Leading (line-height)

Don't forget to set an ample leading for running text! Done using the `line-height` property and you can use a unit-less number (e.g. 1.5 which acts as a multiplier of the font size);

Leading spans from baseline to baseline... and I need some more text to illustrate the point. ☺

# Leading (line-height)

Don't forget to set an ample leading for running text! Done using the `line-height` property and you can use a unit-less number (e.g. 1.5 which acts as a multiplier of the font size);



# Hanging punctuation

Traditionally punctuation marks, bullets, lists, hyphens, and brackets all hang in the margin.

# Hanging punctuation

Traditionally punctuation marks, bullets, lists, hyphens, and brackets all hang in the margin.

“Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Integer a odio. Vivamus placerat felis id risus. Cras mollis est. Etiam mollis vulputate (lorem nullam turpis non massa rhoncus sodales.)”

# Hanging punctuation

Traditionally punctuation marks, bullets, lists, hyphens, and brackets all hang in the margin.

“Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Integer a odio. Vivamus placerat felis id risus. Cras mollis est. Etiam mollis vulputate (lorem nullam turpis non massa rhoncus sodales.)”

## Hanging punctuation (cont'd)

Simply with lists by setting their margins to zero.

## Hanging punctuation (cont'd)

Simply with lists by setting their margins to zero.

Possible for opening punctuation marks using  
background-image or a negative text-indent.

css3 will hopefully support hanging punctuation  
with a proposed hanging-punctuation property.

## Hanging punctuation (cont'd)

Simply with lists by setting their margins to zero.

Possible for opening punctuation marks using  
background-image or a negative text-indent.

css3 will hopefully support hanging punctuation  
with a proposed hanging-punctuation property.

See: <http://tinyurl.com/w3c-hanging-punct>.

## Web: text = UI

Distinguish content elements from application controls (e.g. a user's username and description from a logout link or text-field title).

## Web: text = UI

Distinguish content elements from application controls (e.g. a user's username and description from a logout link or text-field title).

Often done with colour (e.g. setting controls gray), size (making them a tad smaller) and with different typefaces or font styles.

## My Projects

### ■ Project: MadHatter

Client:

Looking Glass Solutions

Description:

A project/client management system.

Latest:

-  Updated theme buttons (ie submit) (by Evan L. on 2008-03-23)
-  Added [Fwd: Madhatter – renaming project breaks link to wiki] (by Evan L. on 2008-03-18)
-  Added IE Compatibility (by Evan L. on 2008-03-11)
-  Updated Chao Yan (by Evan L. on 2008-03-11)
-  Added Add user, contact is a mess (by Andy W. on 2008-02-28)

Staff:

Andy W. (Developer), Pascal K. (Graphic Designer), Evan L. (Designer / Developer), Chao Y. (Support)

## My Projects

### ■ Project: MadHatter

Client: Looking Glass Solutions

Description: A project/client management system.

Latest:  [Updated theme buttons \(ie submit\)](#) (by Evan L. on 2008-03-23)

 [Added \[Fwd: Madhatter – renaming project breaks link to wiki\]](#) (by Evan L. on 2008-03-18)

 [Added IE Compatibility](#) (by Evan L. on 2008-03-11)

 [Updated Chao Yan](#) (by Evan L. on 2008-03-11)

 [Added Add user, contact is a mess](#) (by Andy W. on 2008-02-28)

Staff: [Andy W.](#) (Developer), [Pascal K.](#) (Graphic Designer), [Evan L.](#) (Designer / Developer), [Chao Y.](#) (Support)

---



# Page (grid)

# Page (grid)

Grids bring order to your page and help specify where things should go.

# Page (grid)

Grids bring order to your page and help specify where things should go.

Remember to put in apt “gutters” (margins between columns) to separate your columns.

# Page (grid)

Grids bring order to your page and help specify where things should go.

Remember to put in apt “gutters” (margins between columns) to separate your columns.

Ruler guides help enormously: use the **Web Developer** extension for Firefox or ruler background images (also see the YUI library).

# Subtraction

 Version 7.0  
Khoi Vinh's Web Site

[Home](#) [Archives](#) [Elsewhere](#) [About](#) [Previous](#) [Next](#)

Fri 31 Dec 2004 **Grid Computing... and Design**

Posted 07:51 PM  
Author Khoi Vinh  
Categories [Design](#), [Subtraction](#), [Web Design](#)

Body

The layout grid I used for Subtraction Six.5 was improvised and inconsistent — I hobbled it together without much consideration or foresight, more interested in getting something finished than building something that would continue to make sense as I got more and more serious about the writing I post here. Over time, by virtue of repeated use, I became increasingly and lamentably invested in its tremendous shortcomings. When you make fairly liberal use of illustrations in your posts, you essentially wed yourself to the particulars of the CSS you've established, creating graphics of a certain width or ordering content in a particular method. It works in the short term, but it presents problems when you sit down to redesign.

**Search via Google**

**Quick Access**  
1272 posts since July 2000.

Date

Categories

**Recent Posts**

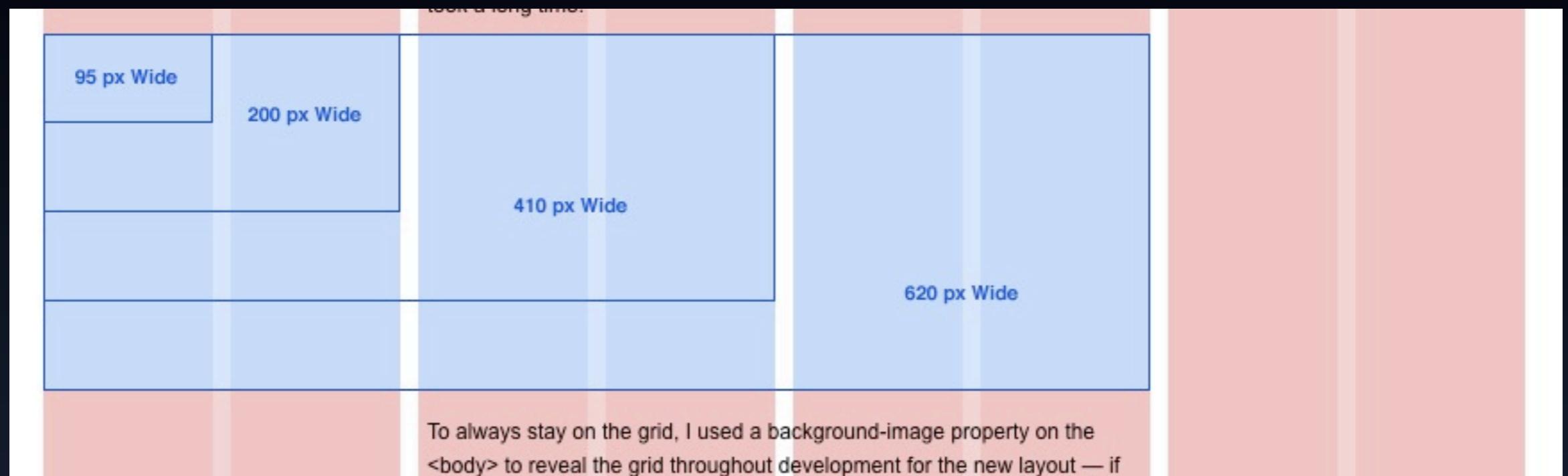
21 Apr 2008 [Talk to the Hand...That Art Directs](#)  
Ask me questions about NYTimes.com

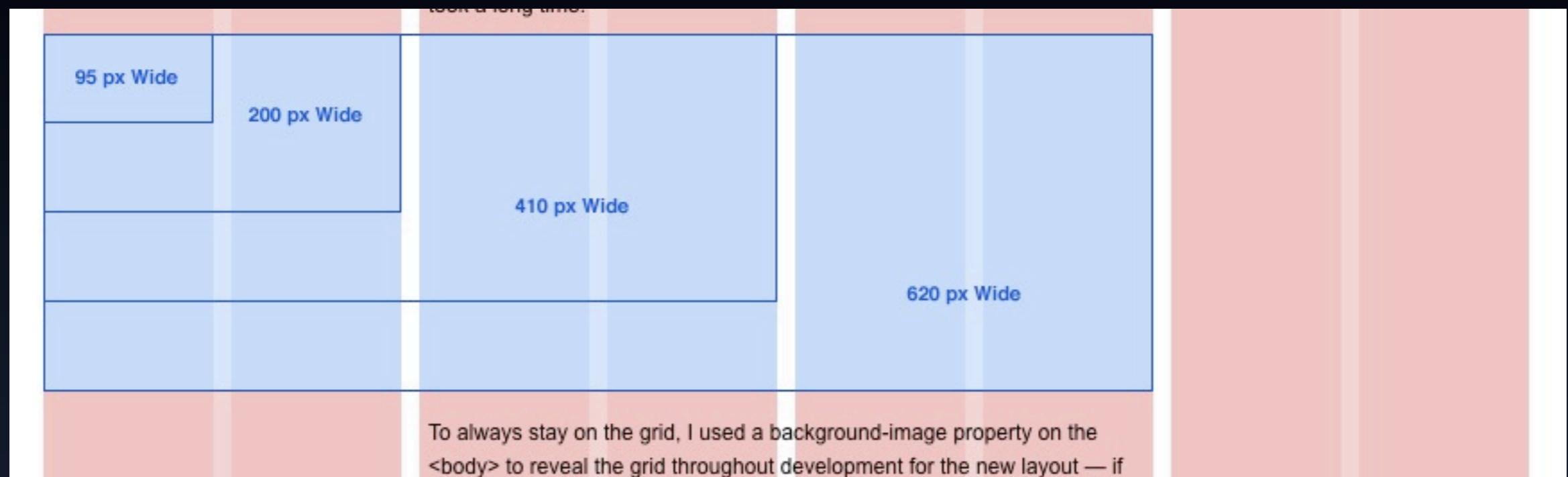
08 Apr 2008 [Notes on Note-keeping Software](#)  
Episode three of my search for an all-purpose memory aid.

07 Apr 2008 [Online Apps Turn Me Offline](#)  
Is software better when it lives on the desktop or in the browser?

## Setting standards-friendly web type

---





Btw: <http://gridulator.com>.



# Addendum

# Addendum

*Please don't use Comic Sans unless you have a  
very, very good reason.*

The Elements  
of Typographic Style  
*version 3.1*  
Robert Bringhurst

ISBN: 0-88179-206-3

# *Inspirational typophiles*

- Cameron Moll: [cameronmoll.com](http://cameronmoll.com)
- Jeff Croft: [jeffcroft.com](http://jeffcroft.com)
- Jeffrey Zeldman: [zeldman.com](http://zeldman.com)
- Mark Boulton: [markboulton.co.uk](http://markboulton.co.uk)
- Richard Rutter: [clagnut.com](http://clagnut.com)
- Jon Tan: [jontangerine.com](http://jontangerine.com)

# Resources & reading

- <http://webtypography.net>
- <http://alistapart.com/topics/design/typography>
- <http://usabletype.org>
- <http://ilovetypography.com>
- <http://www.papress.com/other/thinkingwithtype/>

# Thanks!



klepas@klepas.org



klepas.org



@klepas

---

Licensed CC BY-NC-ND

<http://creativecommons.org/licenses/by-nc-nd/2.5/au/>