



## Article

[Home](#) » [Design and Layout](#) » [Design Principles](#) » Page 1

# The Anatomy of Web Fonts

**By Andy Hume**  
**December 9th 2005**  
**Reader Rating: 8.7**

**If one aspect of design has suffered most in its transition to the Web, it is the art of typography. For years, Web typography involved little more than choosing a typeface and font size. Unstyled Times New Roman was the norm, and the integration of established typographical techniques and rules was unimagined.**

But times change. Since the adoption of [CSS \[1\]](#) into mainstream Web design, we have entered a new age of Web typography. This facet of design has been opened up to the Web designer in a way that wasn't possible in the past. We now have the tools to return typography to its true role within the sphere of design.

However, if we've learnt anything over the last ten years, it's that the Web is not print. The basic principles of typography will cross over to any format, but it's vital that we allow the features of the medium -- in this case, the computer screen or monitor on which our type is displayed -- to define the rules and techniques that we practice.

So, how can we transfer the established and time-tested principles of typography to the online environment? The best place to start is to look at the differences between print and screen, and to understand how those differences will affect our use of typography in Web design.

## The Technology

The most obvious difference between the Web and its print ancestor -- a most important point for those wanting to grasp the art of Web typography -- is that your type is displayed on a computer screen or monitor. If you're reading this tutorial on your computer screen, you're having a very different experience than if you'd printed this text with a modern laser printer.



### Andy Hume



Andy lives in London and works as a Web developer at [Multimap.com](#),

with a focus on Web standards, accessibility, and usability. He writes about Web development, design, and typography at [UsableType.com](#).

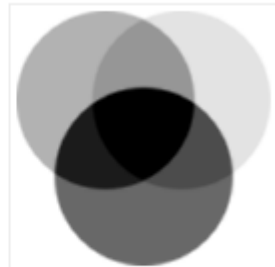
### ***Additive and Subtractive Color Systems***

When you look at a computer screen, you see colors and shapes that are made up of light. Colors are built up by combining red, green, and blue light together in different mixtures and intensities. This is the additive color system.

On the other hand, if you printed this page out, and are reading it on paper, the colors and shapes you're looking at are comprised of pigments in the ink. These colors are perceived by the absorption and reflection of light from their surfaces. This is known as the subtractive color system.



ADDITIVE



SUBTRACTIVE

Why is this important? Well, let's consider the most common color combination for text and its background: black on white. In a subtractive color system (i.e. print), this is a perfectly suitable practice. The contrast of black on white is as stark and clear as possible, making for good legibility and comfortable reading. However, with an additive color system (i.e. on screen), the color white is produced by mixing red, green, and blue at full intensity. This is why the black on white combination can be overly luminous and too harsh on the eyes to allow extended reading on screen. There is never more light radiating from a screen than when it displays pure white, and this intensity can affect the clarity of fine detail in typefaces and other intricate patterns.

Wikipedia gives more details on the [additive](#) [2] and [subtractive](#) [3] color systems.

### ***Screen Resolution***

If you have printed this text with a modern laser printer, you may be enjoying a print resolution of up to 2400 dpi (dots per inch). On screen, you'll be lucky to have a resolution of 96 dpi. The conclusion to this point should be fairly obvious: there's no way you're going to get the same clarity and detail when displaying text on screen as you can with print. It is this factor, more than any other, that defines the recommendations and principles behind good Web typography.

### ***The Unknowns***

The other key issue we should think about is what I like to call the unknowns. One of the biggest hurdles designers have to overcome as they move from print to Web is the idea that they don't have real control over their finished product. No matter how much we try to second-guess users' browsing habits, hardware configuration, and

software choices, we must live with the fact that we cannot control 100% how people experience our Web pages.

In terms of Web typography, this has some pretty sobering consequences. We can spend hours choosing the perfect combination of typefaces to complement our design or meet corporate requirements, but if the user has stipulated that she wants all text to be displayed in 18px Comic Sans, there is absolutely nothing we can do about it.

The key is to think of this not as a limitation of the Web, but as a strength. Which other medium gives so much freedom to the end user? As designers, we merely suggest a design or layout; the final say is in the hands of the individual, and their requirements and choices trump ours every time. Don't inhibit this freedom by assuming users' requirements or by attempting to force too many of your own preferences onto users.

Another unknown that applies particularly to Web typography is anti-aliasing. Modern operating systems such as Windows XP and Mac OS 10 give users the option to display anti-aliased text as standard. This is a good thing, and it makes reading and working on your machine much more comfortable and pleasing to the eye. However, there is a danger when experimenting with typefaces and sizes that we forget that users may not have anti-aliasing available or switched on.



*Comparing anti-aliased and aliased text.*

Again, don't assume anything about the people who use your site. I would encourage any designer to at least check a site's pages with anti-aliasing switched off. The discovery you make may come as a shock, and in the worst case, you may even be moved to rethink your typographic choices.

## **Basic Principles of Web Typography**

---

Having seen the technological issues that are going to inform our typographical choices, it's time for a more detailed look at the principles of typography as they apply to the Web.

### ***Typefaces***

Choosing a typeface is the starting point of typographic design, and may even go on to inform the overall design or feel of a page. Traditionally, Web typography suffers from a lack of options when it comes to typeface choice simply because fonts can be displayed only if they're available on users' computers. The first job of the Web typographer, then, is to distinguish the fonts that are readily available, and to understand which are best suited to his or her particular tasks.

Very generally speaking, there are two main groups of typefaces that are used on the Web: serif and sans-serif.

The difference is easily demonstrated in this image:

# Georgia Verdana

*Serif and sans-serif type*

Serifs are the extra lines or small decorations added to the ends of the main strokes of the Georgia typeface above. The theory behind these serifs is that they help the letters flow, and lead the eye across the text during reading. Serif fonts are very popular in print, and although there is a certain amount of debate regarding which family of typeface is most legible on the screen, I fall firmly in to the camp that believes that sans-serif faces are a more suitable option.

The variable boldness and fine extra strokes of the serif fonts, particularly at smaller sizes of body text, often appear pixilated and untidy. This is still the case even with the most modern anti-aliasing techniques. With anti-aliasing enabled, the serif fonts look blurred (which is exactly what they are) around their curves and terminals. On the other hand, the straight, low contrast, open strokes of a sans-serif font, such as Verdana, will always leave a good impression on-screen.

## ***The Common Web Fonts***

Most designers are probably familiar with what could be considered the common Web fonts. The following CSS rule probably won't raise too many eyebrows.

```
body {  
  font-family: verdana, "trebuchet MS", helvetica, sans-serif;  
}
```

Here, we have three classic fonts that are used all over the Web. However, instead of just copying and pasting this rule, let's look at why we have made the decision to use those three fonts, and what characterizes them as suitable for our purposes.

Verdana is our first choice font here. This font was designed for the screen and is the most common sans-serif typeface used on the Web today.

WIDE LETTER SPACING  
**Verdana**  
WIDE PUNCH WIDTH

*The Verdana face in detail*

Note the generous amount of space between each character, as well as the amount of whitespace within the characters (glyphs) themselves. This is what makes Verdana so legible on screen, and an excellent choice for a

sans-serif on the Web.

Trebuchet is another face created for the screen. Designed in 1996 by Vincent Connare, it is probably the most distinctive of the common Web fonts, and can convey a great deal of energy and personality. Certain features of this typeface depart significantly from what we would expect from a classic grotesk font, for example, the uppercase M seen below.

CLEAR DISTINCT LETTER SHAPES  
**Trebuchet MS**  
WIDE PUNCH WIDTH

*The Trebuchet face in detail*

Despite its distinct personality, Trebuchet's strokes are blocky and clear. It has a large x-height, helping to increase legibility at smaller sizes.

Finally, we've gone for Helvetica. Helvetica, designed in the 1950s and hugely popular throughout the second half of the 20th Century, is a classic in its own right. That it transfers so well to screen is both a testament to its legibility and a blessing for Web typographers.

**Helvetica** TALL X-HEIGHT

*The Helvetica face in detail*

Despite its compact width, Helvetica reads well on screen due to its large x-height. It has a consistent and uniform feel to it, which not only makes it economical with space, but also means it's easy on the eye.

Finally our CSS rule declares sans-serif as a generic font family. This ensures that users' systems will at least default to a sans-serif font if they do not have any of the others we have specified.

If you use the rule above, you'll be certain to provide good, legible text to the vast majority of your users. However, why not experiment a bit more? What are some of the other good screen fonts out there that we can provide to users who have them available?

### ***What Characterizes Good Screen Fonts?***

Having looked at the properties of the above fonts in a bit more detail, you should have a clear idea of the features that define a good screen font:

Low contrast and simple strokes with a consistent weight and thickness

Generous x-height

Generous width and letter spacing

Generous punch width (space within letters)



With those points in mind, you should be in a position to begin looking beyond the common Web fonts. So, what are the design principles you should bear in mind when thinking about typeface choice?

Choose a typeface or a group of faces that will honor and elucidate the character of the text.

- R. Bringhurst

Your typeface sets the tone or personality of your text. Type is a part of your design that the user has no choice but to interact with directly, so its power should be taken seriously. If you were designing a site for a large bank or mortgage company, it's unlikely that you would choose to set the body text in Comic Sans. Comic Sans is a fun, cartoony, frivolous typeface that would not at all be appropriate for the corporate image required.

This is a rather over-the-top example, but the principle is clear: let the nature of your content and the goals of your design lead you in your choice of typeface. This means reading the text and having an understanding of what it conveys, then choosing a typeface that best illustrates and clarifies the key concepts.

The best way to 'get into type' is to immerse yourself in what's available and understand how people are using those fonts. Visit some of the [good type foundries](#) [4], in particular [Linotype](#) [5] and, for the more adventurous, [Letterhead Fonts](#) [6].

Another resource I find invaluable in the early stages of typographic work is the fantastic tool, [Typetester](#) [7]. Typetester allows you to easily style text with any number of CSS properties and compare them to one another on-screen. It even builds the CSS rule for you when you find a combination of styles that you like! This tool is well worth spending some time with, particularly if you're new to many of the detailed text properties in CSS.

This brings us to look a little more closely at those detailed CSS properties. With so many capabilities now available to the Web designer through CSS, how should we be using these properties to deliver the best possible experience to our users?

### ***Size***

Type size is one of the easiest factors to control on-screen, and as such it is often not given the importance that it deserves in the field of Web typography. Size is an important device for giving your content a hierarchy, and the relative sizes of your headings, body text, and footers can have a big influence on the overall feel of a page. Type size is also very closely linked to other characteristics of your page, such as column width, line-height, and so on.

As early as the sixteenth century, typographers had begun to use a common scale for type size, and there is no reason not to replicate this approach on the Web, particularly if you're looking for a traditional and highly legible result. Like a musical scale, the series below has a natural elegance to it, and is considered by typographers to be a standard for relative type sizes.



A great example of a site that uses this type size scale on the Web is Jaredigital.com[<http://www.jaredigital.com>] It's clear that traditional and stylish typography was required for this site, and a little discreet poking around in the CSS finds the rules that make it all possible:

```
h3 {  
  font-size:16px;  
}
```

```
h4 {  
  font-size:12px;  
}
```

```
p {  
  font-size:11px;  
}
```

A good rule of thumb is that the closer to one another are the font sizes of the various levels, the more elegant the overall impression will be.

If you're after a disjointed or modern feel, try using font sizes that are further apart on the scale. A good example of this can be seen at [Coudal.com](http://Coudal.com) [8]. The large headers used here are fairly unusual, and in my opinion, give a much more modern, less traditional feel to the design. Note, however, that the headings are displayed in a much lighter color than the main body text. Another good rule of thumb states that, if you want to use larger sizes for headings, use a lighter color to keep the overall feel of the page balanced. It's also worth pointing out those headings as a particularly successful use of a serif font (in this case, Times) on the screen.

### ***Emphasis***

Emphasizing text is a relatively simple way to bring your words to life on the Web. There is a great fashion at the moment -- particularly in online promotional or advertising text -- to scatter bold font weights liberally around the page. Although traditional typographic convention would warn against the over-use of related faces such as bold or italics, on the Web, this technique can be used to encourage the eye through the text when there are so many other distractions on screen. There are various ways we can introduce emphasis into text:

**Bold:** On the Web, the most common and effective method is the use of a bold face from the current font family. Of course, as with all these techniques, ensure that you do not over-use the bold face. The key is to use bold faces sparingly, or not at all.

**Italics:** Italic text can suffer on low-resolution monitors because of the slanted and more curved shapes of italicized letters. These are likely to look too pixilated when aliased, and too blurred when anti-aliased, and will spoil legibility when used in lengthy body text. However, this should not stop you from using an italic face when convention approves it, for example, quoting foreign words and phrases, or listing books and periodical titles.

**Underlining:** It's commonly understood that underlining text on the Web is not a good idea. Web conventions tell us that any underlined text is a link, and can be clicked on. It's a pretty fair bet that if you go around underlining text that's not clickable, you're going to confuse users.

**Color:** Using color for emphasis is also a rather tricky business. Like underlined words, colored words could be mistaken for a link within body text. In the past it was common to use a distinct background color to give emphasis to a passage of text, but even this has now become a convention for identifying links. Furthermore, there are the [accessibility issues](#) [9] associated with using color to display important meaning on the Web, which further add to the conclusion that using color for emphasis is not an ideal method.

**Capital Letters:** Never set passages of text in full or small caps. Not only is doing this considered rather rude and inelegant, but typographically, it is a very poor choice. CONSIDER THE LEGIBILITY OF THIS SENTENCE COMPARED TO THE ONES THAT PRECEDE AND FOLLOW IT. There is no doubt that, without the distinct shape of the words created by the lower case letters, readability is severely inhibited. The example below highlights this clearly. There is no doubt that it is easier to make out the words in the upper half of the sentence due to the more obvious shape of the words.

**WHICH DO YOU FIND EASIER TO READ?**

**Which do you find easier to read?**

*Note how the upper half of the sentence is easier to read than the lower*

However, nice touches can be achieved using full or small caps in the right situation. Try using small caps for abbreviations and acronyms within text, and of course, full caps for acronyms that stand for personal names or place names. This is a detail that is rarely considered on the Web, but that should be simple to achieve with site-wide CSS files.

### ***Line Spacing***

A crucial detail when setting type on the Web is the issue of line spacing, or leading. The vertical distance between lines of body text can make a huge difference to the legibility and overall style of your message. A comparison of the two blocks of text below should demonstrate the point clearly. A general rule of thumb when selecting a line-height value is that the longer the line length (measure) of your text, the more leading is required for legibility.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla viverra gravida eros. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In convallis lorem. Nam massa. Donec diam purus, rhoncus vel, aliquet sit amet, hendrerit in, diam. Nulla adipiscing eleifend orci.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla viverra gravida eros. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In convallis lorem. Nam massa. Donec diam purus, rhoncus vel, aliquet sit amet, hendrerit in, diam. Nulla adipiscing eleifend orci.

*Notice the clarity that the increased line height brings to the text*

The default line-height for most browsers is around 1.2 (where 1 is where the tops of one row of characters would touch the base of the row above). This is not sufficient for text on screen; I'd recommend increasing your line-height value to between 1.4 and 1.6.

### ***Letter Spacing and Kerning***

In my opinion, letter spacing (or tracking) as controlled by the letter-spacing property in CSS should never be used for body text. It's not very cross-browser friendly, and what you may see in [Firefox](#) [10] may be drastically different in Internet Explorer or Safari. Furthermore, it seems wrong to mess about with the tracking that each font has built into it. The type designer carefully calculates these spaces, and most of the time, they're going to provide good results without any extra tampering.

However, letter spacing can be a very effective tool in short headings. A classic example of this can be seen at [Zeldman.com](#) [11]. The large letter spacing of the titles in the right hand column allows for a bolder, uppercase typeface, but ensures that the overall balance of the page is consistent.



*Letter spacing of titles at Zeldman.com*

Kerning refers to improving the appearance of a word by adjusting spacing between certain pairs of letters. On the Web, it's not realistic to attempt this with body text. However, if you're creating headers or logos in graphics

software, a little kerning can make a dramatic difference to the overall result. Look at the two examples below. The first shows the default kerning of the Verdana font; the second has been tweaked in Fireworks with the goal of creating a more balanced and well-proportioned word shape.

### ***Word Spacing***

As with letter spacing, word spacing should be considered carefully before it's applied to the screen. I am of the belief that if you think your text needs spacing and tracking manually, then you have probably chosen the wrong typeface for the job. As with all the text properties that CSS allows us to control, consistency is the key. Keep readers focused on the text, and make it easy for their eyes to flow through the words. If you're breaking up the letters and words, you'll only serve to distract them, and disrupt the flow of the text.

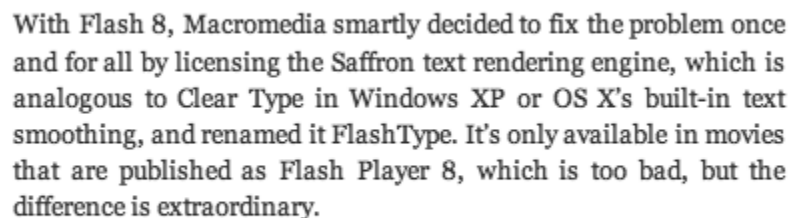
### ***Alignment***

Margins can have a surprisingly large influence on the look and feel of your pages. They help keep your text apart from other elements of your site, and from the edges of the browser window. Larger margins produce an open, free feeling, while smaller, tighter margins give a constricted and more intimate aesthetic.

Assuming your content is in a language that flows left-to-right (ltr), you should nearly always left-align your text on the Web. Right-aligning should only be used for very specific purposes, and certainly not for long passages of text. The ragged left edge of right-aligned text makes it very difficult for the eye to move from one line to the next and reduces legibility considerably.

Text that is set flush with both left and right margins is known as justified text. Justification is extremely popular in books and newspapers, and gives a uniform and controlled feel to a page. Justification on the Web would be far more popular if it weren't for the fact that no browser or operating system has a hyphenation dictionary built in. This means that the only way the browser has to justify text is to space out and/or squash up words to fit into the specific line length. This is fine to an extent, but there comes a point -- particularly with shorter line lengths -- when a bit more flexibility is required.

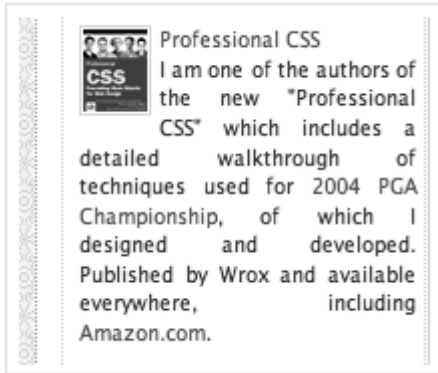
The [designer Todd Dominey](#) [12] provides an option on his site to justify all text, and the two screen shots below shows the good and bad results of doing that.



With Flash 8, Macromedia smartly decided to fix the problem once and for all by licensing the Saffron text rendering engine, which is analogous to Clear Type in Windows XP or OS X's built-in text smoothing, and renamed it FlashType. It's only available in movies that are published as Flash Player 8, which is too bad, but the difference is extraordinary.

To see what I mean, check out this comparison. The image on the

*Good justified text at Whatdoiknow.org*



*Bad justified text at Whatdoiknow.org*

The text in the wider column looks great, and the browser has dealt well with the justification. However, in the smaller left-hand column you can see that the word spacing is very mixed and inconsistent, and this drastically affects the legibility of the text.

## Extending Web Typography

---

While a lot can be achieved with CSS in the ways described above, designers must still endure a number of limitations when it comes to Web typography. We are still stuck with a limited number of good screen fonts, but even if more were available, we'd still be relying on users having them available on their machines. With this system, designers are never going to have the true control of type that they desire. It is frustrating that in the ten years since designers came to the Web, this problem has not been solved.

### ***Replacing Text***

One way in which designers work around the limitations today is to use image and [Flash](#) [13] text replacement techniques to display headers or navigation. This kind of text is often an integral part of a site's design or brand, and designers are keen to get as much control over these elements as possible.

The rules for using images or Flash to replace text are no different from those we use to choose body text typefaces. Legibility should always be your prime concern, followed by suitability for the task at hand. However, you do have a little more flexibility, mainly because you are more likely to be working with large font sizes and thus many legibility issues are already solved. This situation may then be a suitable case for experimentation, and for basing a typeface choice solely on the character, or personality that you're trying to convey in your design.

One Flash replacement method that has caused a lot of buzz over the last year or so is [sIFR](#) [14] (pronounced siffer). SIFR is touted as an [accessible](#) [15], scalable method of embedding custom typefaces in Web pages, and as such has generated a great deal of excitement among designers.

It is a superb tool for Web designers to have at their disposal. There are specific cases in which sIFR is an unrivalled solution to the problem of creating dynamic headers in a custom typeface. However, it is not the final solution to custom typography on the Web, and it's important to understand what the specific cases are before

deploying it in your next design. Like any powerful tool, sIFR has the potential for serious misuse. To learn more about sIFR and its use cases, visit [the official sIFR Wiki](#) [16].

## The Future

---

Flash replacement techniques, such as sIFR, hint at the future of Web typography, but they themselves are not the future. They are stop-gap solutions that, with luck, will show browser makers, type foundries, and the [W3C](#) [17] that Web designers will not rest until they have a way to display custom typefaces in their pages.

There have been attempts in the past to solve this problem, notably [Microsoft's Web Embedding Fonts Tool \(WEFT\)](#) [18], which allows for fonts to be stored on a Web server and downloaded when required. Unfortunately, this has never developed into a satisfactory tool, simply because no other browser maker has implemented the Microsoft technology. The type foundries too, are worried about the consequences of having their copyrighted type easily spread around the Web.

Once this problem is finally overcome, which it surely will be (albeit later, rather than sooner), we will enter another age of Web typography. However, the same rules will still apply, because the fundamental principles of typography will transfer to any medium. The real challenge is to continue understanding the medium we are designing for as it develops, and to continue adapting these rules and guidelines to fit it.

In the meantime, though, keep an eye on what will surely prove to be an incredible new resource for Web typographers: [Elements of Typographic Style Applied to the Web](#) [19].

This new site takes the principles of Robert Bringhurst's classic book, [The Elements of Typographic Style](#) [20], and steps through them one by one, explaining how they can be applied to the Web. If this article has got you thinking about what you could do with typography on your site, this will be the perfect place to take your knowledge a step further.

The site's owner, Richard Rutter, is adding one chapter per week. Each illustrates examples of practical CSS that can be used to replicate the principles of Bringhurst's classic text, and explaining how and why they work on the Web. It's definitely a site to keep an eye on over the coming months.

[Back to SitePoint.com](#)

[1] [/glossary.php?q=C#term\\_8](#)

[2] [http://en.wikipedia.org/wiki/Additive\\_color](http://en.wikipedia.org/wiki/Additive_color)

[3] [http://en.wikipedia.org/wiki/Subtractive\\_color](http://en.wikipedia.org/wiki/Subtractive_color)

[4] <http://www.typebase.com/foundries.html>

[5] <http://www.linotype.com/>

[6] <http://www.letterheadfonts.com/>

[7] <http://typetester.maratz.com/>

[8] <http://www.coudal.com>

[9] <http://www.w3.org/TR/WAI-WEBCONTENT/#gl-color>

- [10] /glossary.php?q=F#term\_45
- [11] <http://www.zeldman.com>
- [12] <http://www.whatdoiknow.org>
- [13] /glossary.php?q=F#term\_16
- [14] <http://www.mikeindustries.com/sifr/>
- [15] /glossary.php?q=A#term\_61
- [16] <http://wiki.novemberborn.net/sifr/>
- [17] /glossary.php?q=W#term\_49
- [18] <http://www.microsoft.com/typography/web/embedding/weft3/default.htm>
- [19] <http://webtypography.net>
- [20] <http://www.amazon.com/gp/product/0881791326/>

» Page 1