

# Typographic Style for Computer Scientists

Geoffrey Washburn

Draft produced on August 12, 2009

## 1 Introduction

Much of the information in this article was gleaned from a study of several primary sources:

- *Mathematical Writing* by Donald E. Knuth, Tracy Larrabee, and Paul M. Roberts (1989). Most of this book can actually be downloaded from Knuth's website.
- *The Elements of Typographic Style* by Robert Bringhurst (2004).
- *Stop Stealing Sheep & find out how type works* by Erik Spiekermann and E.M. Ginger (2003).
- *The Visual Display of Quantitative Information* by Edward R. Tufte (2001).
- *The Chicago Manual of Style* (2003).

If you have the time, I would suggest reading them directly.

Also note that the document is written from the perspective of someone working in programming languages and type systems. It is my hope that despite the choice of specific examples, the ideas presented should remain valid for most of computer science, and even those working in mathematics.

## 2 Common “mistakes”

### 2.1 Mixing mathematics and prose

The following items are adapted from Knuth, Larrabee, and Roberts (1989).

pages 1 – 6

1. Symbols in different formulas must be separated by words.

**Bad:** Consider  $S_q, q < p$ .

**Good:** Consider  $S_q$ , where  $q < p$ .

2. Do not start a sentence with a symbol.
3. Unless you are referring to the actual “logical operators”, do not use the symbols  $\therefore, \Rightarrow, \forall, \exists, \in$ , replace them by their corresponding words.
4. The statement just preceding a theorem, algorithm, etc. should either be a complete sentence or end with a colon.
5. Display important formulas on lines by themselves.
6. Small numbers should be spelled out when used as adjectives, but not when referring to the mathematical objects.
7. The text preceding displayed equations does not use any special punctuation, in particular no colon. Note that displayed equations are different than theorems and algorithms as described in Item 4.
8. One rule that I have not yet mentioned, and only seems to be implicitly described by Knuth et al., is that if the displayed equation is immediately followed with punctuation, that punctuation should go in the displayed equation. So

**Bad:** Our model is based on the equation

...an equation

. Here ... is ....

**Good:** Our model is based on the equation

...an equation.

Here ... is ....

and

**Bad:** Therefore, we can conclude

...an equation

, so that ... is ....

**Good:** Therefore, we can conclude

...an equation,

so that ... is ....

9. Avoid nesting subscripts, i.e.  $x_{i_j}$ .

## 2.2 Boxes and vertical lines

According to Spiekermann and Ginger (2003):

A sure way to improve the look and function of any information-intensive document is to eliminate boxes. Vertical lines are almost always unnecessary. Type creates its own vertical divisions along the lefthand edges of columns as long as there is sufficient space between columns. A vertical line is wasteful because it need precious space on either side. Use space to divide elements from each other. Utilize horizontal lines to accentuate areas of the page. The edge of the paper makes its own box and doesn't require more boxes inside it.

page 153

Bringhurst says (2004):

A rule located at the edge of a table, separating the first or final column from the adjacent empty space, ordinarily serves no function.

page 71

Finally, one should keep in mind Tufte's principle of maximizing *data-ink*. That is maximizing the amount of "ink" used to present information in proportion to "ink" that does not present any actual information (2001).

Chapter 6

## 2.3 Small caps

Bringhurst says:

For abbreviations and acronyms in the midst of normal text, use spaced small caps.

page 48

He goes on to say:

This is a good rule for everything except two-letter geographical acronym and acronyms that stand for personal names. Thus: 3:00AM 3:00PM, the ninth century CE, 450 BC to AS 450, the OAS and NATO; World War II or WWII; but JFK and Fr J.A.S O'Brien, OMI; HMS *Hyphothesis* and USS *Ticonderoga*; Washington, DC, and Mexico, DF; J.S. Bach's Prelude and Fugue in B♭ minor, BMV 867.

Many typographers prefer to use small caps for postal abbreviations (San Francisco, CA 94119), and for geographic acronyms longer than two letters. Thus, the USA, or in Spanish, *los EEUU*, and Sydney, NSW. But the need for consistency intervenes when long and short abbreviations fall together. From the viewpoint of the typographer, small caps are preferable in faces with fine features and small x-height, full caps in faces with large x-height and robust form.

However, these rules do not explain how small caps interact with the usual rules for capitalization. The most common case where it arises is when a sentence starts with text that should be in small caps as described by the rules above. In correspondence with Bringhurst(2006), he suggests a preference towards keeping the text in small caps:

I have no objection myself to starting a sentence with small caps. The reasons for this are (1) I think it looks better than any of the alternatives, and (2) the best typographers of the Cinquecento did it. If I could improve on their practice in this respect, I would happily do so, but I can't.

I also don't object to capitalizing a word in small caps at the beginning of a sentence, *if it is indeed a word* instead of an acronym. (NATO, for this purpose is a word, because we pronounce it as such; IBM is not, because we pronounce it as a string of letters. If you want to begin a sentence with NATO rather than NATO, I won't complain; but I would complain if your sentence began with Ibm instead of IBM or with OECD instead of OECD. Under these conditions, the only obvious reason to prefer NATO to NATO at the beginning of a sentence is consistency between verbal and literal acronyms.)

Small caps can be used in L<sup>A</sup>T<sub>E</sub>X with the `\textsc` command. However be sure to remember that inside of `\textsc` you only get small caps for lowercase letters. That is `\textsc{AM}` gives AM while `\textsc{am}` gives AM.

## 2.4 Old-style numbers

There are two kinds of “numbers” or *figures*: lining or titling numbers and text numbers. The latter are sometimes referred to as “old-style numbers”. Bringhurst says:

Use titling figures with full caps, and text figures in all other circumstances.

page 46

He then goes on to say:

So the date is 23 August 1832; it could be 3:00 AM in Apartment 6-B,  
213-A Beacon Hill; it is 27°C or 81°F;

...

BUT IT IS 1832 AND 81° IN FULL CAPITALS.

Small caps text is considered to be “lowercase” and should be used with text figures.

In  $\LaTeX$  titling numbers are default. The easiest way to use text figures is to use the `\oldstylenums` command. Individual text figures can be accessed via `\textzerooldstyle` through `\textnineoldstyle`. One could conceivably make them the default by adjusting the  $\TeX$  character codes.

If you are using  $X_{\LaTeX}$  with the `fontspec` package, it is straightforward to configure text figures to be the default in your document. If you would like your document to be typeset in Palatino with text figures as the default you would just place `\setmainfont [Numbers=OldStyle] {Palatino}` in the document’s preamble. To switch back to titling numbers within a particular scope, use `\addfontfeatures {Numbers={Lining}}`.

## 3 The semantics use of typefaces

Most readers will be familiar with the practice of specific typefaces to give text a specific meaning. For example, depending on context the use of an italicized typeface could indicate emphasis or that the text is a book or journal. In computer science writing, as in any discipline, there are additional semantic conventions that we tend to employ.

Perhaps the most common example of giving a typeface semantic meaning in computer science is the use of a monospace or “typewriter font” to indicate source code or specific input or output to a computer. This tradition no doubt evolved from the fact that, to this day, most terminals and text editors default to the use of a monospace font.

Another example of the semantic use of typefaces is the long standing tradition in mathematics and computer science of letters, but not numbers, in mathematical statements generally being typeset in italics. This tradition has no doubt been greatly reinforced by the widespread use of  $\text{\TeX}$ , where italic math is the default.

While the use of a monospace typeface for code and italics in mathematical statements is traditional, it is certainly acceptable to break with tradition. It is reasonable to see monospaced text as an archaic and Herman Zapf's typeface AMS Euler, which was intended for mathematical text, is an upright rather than italic typeface. Making choices that run against tradition, such as using a monospace typeface for mathematics, can be acceptable but you run the risk of confusing readers for whom these traditions are the norm. Overall, what is important is not the specific choice of typeface, but the fact that a distinction is being made to assist the reader. Consider the differences between the following two versions of the same text fragment:

... let  $z^2$  equal  $x^2 + y^2$  ...

Alternately:

... let  $z^2$  equal  $x^2 + y^2$  ...

My hope is that you will find that the second version scans better and enables you to more quickly pick out mathematics from the surrounding prose. Unfortunately, at this time I am not aware of any specific published research to support this intuition.

Given the propensity for computer scientists to make categorical distinctions, in some texts it is possible, as Bringhurst put it, for this good idea to “be carried to ludicrous extremes” (2006). The key is to strike a balance between giving the reader too little semantic information to easily parse the text and overwhelming them with subtle typographic cues.

As with most typographical trade-offs, it is often best to decide how to map semantic meaning on to typefaces on a per project basis. The most common schema I use in practice is to divide the text into prose, code, and mathematics. If the code or mathematics in the text is particularly complicated, I sometimes then subdivide these categories further.

I also I recommend settling on a convention that makes it easy for the reader to pick out the meta-variables in your code and mathematics. This convention does not necessarily need to involve switching typefaces. For example, it is common in programming languages texts to use Greek, rather than Roman, letters for many of the meta-variables. Regardless, it should be easy to distinguish meta-variables from the surrounding code or mathematics.

As an example, consider the following inference rule adapted from “Featherweight Java: A Minimal Core Calculus for Java and GJ” by Igarashi, Pierce, and Wadler (2001).

$$\frac{\bar{x} : \bar{C}, \text{this} : C \vdash e_0 : E_0 \quad E_0 <: C_0 \quad \text{class } C \text{ extends } D \{ \dots \} \quad \text{if } mtype(m, D) = \bar{D} \rightarrow D_0, \text{ then } \bar{C} = \bar{D} \text{ and } C_0 = D_0}{C_0 \ m(\bar{C} \ \bar{x}) \ \{ \text{return } e_0; \} \ \text{OK IN } C}$$

In this rule all the meta-variables and many parts of the judgments are all in the same monospace typeface as the the concrete syntax of Featherweight Java. With so little visual distinction, you could find this rule a little difficult to read. Now consider this revised version of the same rule.

$$\frac{\bar{x} : \bar{C}, \text{this} : C \vdash e_0 : E_0 \quad E_0 \leq C_0 \quad \text{class } C \text{ extends } D \{ \dots \} \quad \text{if } mtype(m, D) = \bar{D} \rightarrow D_0, \text{ then } \bar{C} = \bar{D} \text{ and } C_0 = D_0}{C_0 \ m(\bar{C} \ \bar{x}) \ \{ \text{return } e_0; \} \ \text{ok} \in C}$$

In this revised rule, meta-variables are in the italicized typeface to which  $\text{T}_{\text{E}}\text{X}_{\text{math}}$  defaults and the judgements use a combination of a regular Roman typeface and symbols. Elements that correspond to the concrete syntax of Featherweight Java are still in a monospace typeface. However, italics are not the only alternative. Another option would be to simply use a different Roman typeface. Here is a version using AMS Euler.

$$\frac{\bar{x} : \bar{C}, \text{this} : C \vdash e_0 : E_0 \quad E_0 \leq C_0 \quad \text{class } C \text{ extends } D \{ \dots \} \quad \text{if } mtype(m, D) = \bar{D} \rightarrow D_0, \text{ then } \bar{C} = \bar{D} \text{ and } C_0 = D_0}{C_0 \ m(\bar{C} \ \bar{x}) \ \{ \text{return } e_0; \} \ \text{ok} \in C}$$

Given your individual aesthetic there many other alternatives that you might consider or prefer, but it is my hope that you find the alternatives I have presented easier to parse than the original.

### 3.1 Color

One option to prevent your code and text from looking too much like a ransom note is to instead use different colors rather than different typefaces. For example, in some of Wadler’s papers he will make some parts of his text red and some parts of the text blue. The the following two inference rules are adapted from his paper “Call-by-Value is Dual to Call-by-Name” (2003).

$$\frac{\Gamma \rightarrow \Theta, A \quad \Gamma \rightarrow \Theta, B}{\Gamma \rightarrow \Theta, A \ \& \ B} \ \&R \quad \frac{\Gamma \rightarrow \Theta \mid M : A \quad \Gamma \rightarrow \Theta \mid N : B}{\Gamma \rightarrow \Theta \mid \langle M, N \rangle : A \ \& \ B} \ \&R$$

The left rule is the original rule as Gentzen would have written it, and the right rule augments it with proof terms. The use of the color red clearly calls out the parts of the rule that are new. However, nowhere in his paper does he explain the intended semantics of color, so the reader is left to infer it. This omission may have been to avoid confusing readers when they are looking at a black and white version of the paper.

While there is already a long tradition in software of coloring syntax based upon its semantics, unfortunately, because of printing costs few journals or conference proceedings are printed in color. If you are writing a book, color printing may be an option but it is still far from common. It is probably easiest to take advantage of color in presentations.

Another drawback to color is that as much as seven to ten percent of the male population (it is significantly rarer in females), suffer from some form of color-blindness (Wikipedia 2009). For example, consider this purposely extreme example:

Normal vision	Deuteranomaly	Protanomaly
$\lambda x:\tau.e$	$\lambda x:\tau.e$	$\lambda x:\tau.e$

Here I have simulated how a semantically colored typed  $\lambda$ -expression would be seen by someone with the two of the most common forms of color blindness, which affect the perception of red and green. I am not sure how often it proves to be a problem for less contrived color schemes, but I have received a complaint during one talk I gave. When trying out a new color scheme it might be useful to try using Vischeck (Dougherty and Wade 2009) to see if it is still comprehensible to a colorblind viewer.

## Acknowledge

Thanks to Donna Malayeri for suggestions and encouragement.

## References

Bringhurst, Robert. 2004. *The elements of typographic style*. version 3.0 edition. Post Office Box 147, Point Roberts, Washington 98281: Hartley & Marks Publishers.

———. 2006. Personal communication.

Dougherty, Bob, and Alex Wade. 2009. Vischeck. <http://www.vischeck.com/>. [Online; accessed 2-August-2009].

- Igarashi, Atsushi, Benjamin C. Pierce, and Philip Wadler. 2001. Featherweight java: a minimal core calculus for java and gj. *ACM Transactions on Programming Languages and Systems* 23(3):396–450.
- Knuth, Donald E., Tracy Larabee, and Paul M. Roberts. 1989. *Mathematical writing*. Post Office Box 90973, Washington, District of Columbia 20036: The Mathematical Association of America. Also available as Stanford University Technical Report STAN-CS-88-1193.
- Spiekermann, Erik, and E.M. Ginger. 2003. *Stop Stealing Sheep & find out how type works*. 2nd edition. 1249 Eighth Street, Berkeley, California 94710: Adobe Press / Peachpit Press.
- The University of Chicago Press, editor. 2003. *The chicago manual of style*. 15th edition edition. Chicago, Illinois: The University of Chicago Press.
- Tufte, Edward R. 2001. *The visual display of quantitative information*. 2nd edition. Post Office Box 430, Cheshire, Connecticut 06410: Graphics Press.
- Wadler, Philip. 2003. Call-by-value is dual to call-by-name. In *ICFP '03: Proceedings of the eighth ACM SIGPLAN international conference on functional programming*, 189–201. Uppsala, Sweden: ACM.
- Wikipedia. 2009. Color blindness. [http://en.wikipedia.org/w/index.php?title=Color\\_blindness&oldid=305554178](http://en.wikipedia.org/w/index.php?title=Color_blindness&oldid=305554178). [Online; accessed 2-August-2009].