The NeXT™ Style Guide

June 29, 1990

Written by Caroline Rose
Trademark issues by Adrienne Wong
Hardware conventions by Catherine Mikkelsen
Book design by Eddie Lee

Contents

The NeXT Style Guide

About This Guide

This guide presents standards and conventions that should be followed in all of NeXT's written materials, both printed and on-line. Although it usually refers to "manuals," "user documentation," and "technical documentation" (meaning documentation for application developers), its scope includes reference manuals, tutorials, course notes, Marketing communications, help from applications, and anything similar that's seen by people outside NeXT. Although oriented toward NeXT employees, it should also be helpful to third-party developers.

The Style Guide presents general philosophy as well as specific examples, many drawn from actual edited drafts. It will continually be updated as the need to set and document new conventions becomes apparent. This version of it is only the beginning, and is known to have "holes." You can help fill them: Please direct any questions or comments to me, Caroline Rose.

General Style

Guideline: In general, the NeXT style is casual and conversational without being too chatty, slangy, or humorous. Information is written as it would be expressed verbally.

Examples: Use second person ("you") or imperative, as appropriate. Use contractions wherever natural (but don't use "it'd," "it'll," "that'd," or "that'll"). Use active voice as much as possible and reasonable. Don't worry about ending a sentence with a preposition if that's what sounds most natural—for example, "the directory you're copying from."

Guideline: Use simple words.

Examples: Don't use "moreover." Use "required" rather than "obligatory." Avoid "via." Use "depends on," not "depends upon" or "is dependent on." Don't use "datum" or "datums" (use "data" for singular as well as plural). Avoid "prior to" and "utilize" (try "before" and "use"). Avoid using "interface," "input," or "output" as a verb. Don't use "monitor" when you can use "screen." Avoid "desire" or "wish" (try "want"). Use "speaker" (not "loudspeaker") and "stereo," but not "mono" (use "monophonic").

Guideline: Don’t use unnecessary words.

Examples: Use “to” rather than “in order to” wherever reasonable. Don’t use “of” after “all,” “inside,” or “outside,” unless needed for sentence rhythm. Use “whether” rather than “whether or not” unless there’s a reason for the “or not” (as in “0 or 1, depending on whether or not”).

Guideline: Avoid idiosyncrasies.

Examples: “Basically,” as in “There are basically two types,” is an individual writer’s habit and rarely adds meaning.

Guideline: Don’t be obscure.

Example: Don’t say “specify” if you mean “type,” or “select” if you mean “click.” Define every acronym the first time you use it, except for those that are so arcane or irrelevant as to hardly be considered acronyms anymore (like BASIC and ASCII).

Guideline: Don’t be sexist.

Example: Don’t use “he” or “she” to mean a person in general. Also avoid “he or she.” Write around it.

Guideline: Unless this guide indicates otherwise, don’t break the “old rules” just because everyone else seems to be doing it.

Example: Don’t use “very unique” or “most unique.” Most important, don’t use “most importantly” at the beginning of a sentence like this one.

Terminology and Usage

User Documentation

This section gives the terminology and usage that’s appropriate in user documentation, and also includes a section that points out some differences between our terminology and the Macintosh® terminology. It’s by no means complete; as another source, see the glossaries in the NeXT user manuals.

Usually these same conventions apply to technical documentation, but in some cases there are differences (see the “Technical Documentation” section below).

Keyboard and Characters

Capitalize key labels only if they actually appear on the key—for example, “Backspace key” and “Power key”, but “brightness keys” and “arrow keys.” (Note we use “arrow keys,” not “direction keys” or “cursor keys.”)

Refer to “the,” not “a,” Shift, Command, or Alternate key (even though there are two of each).

“Press” a key, but “type” a character. “Type” is preferable to “press” when referring to a combination of keys, such as Control-A, but avoid if possible (for example, if the imperative isn’t needed, as in “You can also use Command-f to do this,” or “You can do this with Command-f”).

Don’t use “Return” as a verb meaning “press Return.”

The notation to use when referring to a combination of keys is shown in the table below.

Notation	**Meaning**
Alternate-x	Hold down Alternate while typing lowercase x.
Alternate-X	Hold down Alternate while typing uppercase X (with either Shift or Alpha Lock).
Alternate-Shift-x	Same as Alternate-X.
Command-d	Hold down Command while typing lowercase d; if Alpha Lock is on, pressing the D key will still produce lowercase d when Command is down.
Command-Shift-D	Hold down Command and Shift while pressing the D key. Alpha Lock won’t work for producing uppercase D in this case.
Control-X	Hold down Control while pressing the X key, with or without Shift or Alpha Lock (case doesn’t matter with Control).

Here are some of our conventions for character names:

Character	**Name**
_	“underscore,” not “underbar”
=	“equal sign,” not “equals sign”
space	“space” or “space character” (if necessary for clarity), not “blank”
‘ “ ' "	“quote” or “quotation mark,” depending on the context (“single quotes” or “double quotes” is preferable to “single quotation marks” or “double quotation marks”)
\	“back slash,” not “backslash”
` on keypad	“back quote,” not “backquote” (to show the symbol that appears on the key, you must type Alternate-i, space, and then Delete) *<<Alternate-i by itself suffices in older versions of the software>>*
[ ]	“square brackets” (or just “brackets” if there’s no need to distinguish from angle brackets)
< >	“angle brackets”
{ }	“braces”

Don’t call the back quote key the “tilde key.” Don’t say “type tilde” when you mean “press the back quote key.”

In user documentation, use “capital” rather than “uppercase.”

Mouse

The small graphic that tracks the mouse on the screen is the “cursor,” not the “pointer.”

Use “mouse location,” not “mouse position.”

Point “to” something with the mouse, not “at.” Refer to “a” mouse button, not “the” mouse button (since there are two, normally interchangeable).

Don’t “click in” or “click on” an individual graphic control such as a button or icon, just “click” it. Don’t “press” a button unless you mean holding down the mouse button without releasing it (as in pressing a scroll button to scroll continuously).

"Click in" a general area such as in a window or in text. A blinking "vertical bar" marks the "insertion point" in text (don't call it the "cursor").

Commands and Controls

"Choose," don't "select," a command "from" a menu. Items are "in," not "on," lists, menus, and windows (except for choosing a command "from" a menu). "Command" in our graphical user interface refers only to something in a menu or to its keyboard alternative, not to a button or an item in a "pop-up list" (note it's not called a "pop-up menu").

In procedural steps, here's how to express choosing from a submenu (these examples assume that the Document command is in the application's main menu—and that the details of choosing from a submenu in general have already been explained elsewhere):

> Choose Open from the Document menu.
>
> Choose Manager from the Document menu, then choose Open Ancestors from the Manager menu.

These examples also assume that the reader knows this is in the context of the current application; otherwise use, for example, "from Edit's Document menu."

Regarding the term "pop-up list": It's optional; you may not need to use it, depending on how much they come up in what you're documenting. The 1.0 NeXT user documentation doesn't use the term.

Cross-references to commands that put up panels should be, for example, "You can do this with the Preferences command," not "You can do this in the Preferences panel."

Don't mention pressing Return as the only option in a panel that includes an OK button marked with a Return symbol; mention clicking OK and then, if you want to also mention it, add "(or press Return)."

Buttons that are labeled with icons should not be called "icons." Instead, follow the convention we observe in using the terms "miniaturize button" and "close button": Name it after its function, in lowercase, followed by "button"; don't capitalize the function, because that will it imply it's literally the button label, and the reader will look for it as such.

A button that's part of a pop-up list can't very well be named after whatever the button currently happens to be labeled with. In cases where there is text above the button stating what its function is—for example, "Color:" above a Red/Green/Blue pop-up list—consider that text to be the label. For example, you might say "Use the Color button to set the color," or "Choose one of the following from the Color pop-up list." If there's no text label, name the button or list after its function (for example, WriteNow's "zoom button").

Other User Interface

Use “displayed” for something on the screen, not “printed.”

Use “display” and “highlight” as transitive verbs: Something “is displayed” or “is highlighted,” rather than “displays” or “highlights.”

The past tense of “highlight” is “highlighted,” not “highlit.”

Windows are stacked from “front” to “back,” not from “top” to “bottom.”

Don’t use “grow” or “size” as a transitive verb (when referring, for example, to windows). Windows are “resized.”

Things go “on,” not “in,” the pasteboard.

Use “justified” only when referring to text that’s aligned with both the left and right margins (see *The Chicago Manual of Style*, paragraph 19.26). For text aligned only on the left, for example, you can use “left-aligned,” “flush left,” or “ragged right” (in order of preference).

The user logs “in” (not “on”), and logs “into” (not “in to”) the system. This procedure is referred to as “login.”

In user documentation, “start up” an application rather than “launch” it. Use, for example, “start up WriteNow” rather than “start WriteNow up.”

Operating System

Use “UNIX 4.3BSD,” not “UNIX 4.3 BSD” or “UNIX BSD 4.3.” The first time it’s used in a manual, follow it with “(Berkeley Software Distribution).”

AT&T requires that “UNIX®” not be used as a noun by itself. Wherever possible, follow it with “operating system” or “system.”

Our operating system is “Mach,” not “UNIX” or “Mach UNIX,” but it is “UNIX-compatible.” Where its features overlap with UNIX, however, we call them UNIX features (for example, “UNIX commands”).

Tasks are performed “concurrently,” not “simultaneously,” in a multitasking system like UNIX or Mach.

Miscellaneous

For common abbreviations of multiple-word terms—for example, "dock" for "application dock"—use the full term the first time it's brought up and anytime after that when it's been a long time since you last brought it up (or at a break like the beginning of a chapter). Otherwise, use the abbreviated term.

Use "the NeXT Computer," not "the NeXT." Use "cube" only when you need to refer specifically to that piece of hardware (for example, when talking about the ports on the back of the cube).

Use "NeXT Authorized Service Center" the first time; after that, you can drop the "NeXT."

Use "lower left" (or right) and "upper left," no hyphen. Don't use "left hand" or "right hand."

Use "enclosed in," not "enclosed by."

Use "waked," not "woken," as the past participle of "wake"—for example, "This thread can be waked up by a timeout."

Remember that "or" is singular, not plural. So use "the Hide or Quit command," not "the Hide or Quit commands." Similarly, use "and" instead of "or" in:

> The value should be one of the following: alpha, beta, and gamma.

The use of "or" would be incorrect here because it should be a list of several alternatives, not a single alternative. However, the following is correct and even preferable:

> The value should be alpha, beta, or gamma.

Avoid "If..., then..." Usually, "then" isn't necessary, and it makes the prose read like code.

Use "on the disk" rather than "on disk."

Use "link," not "soft link."

Don't use "app"; spell out "application." Ditto for "AppKit"; it's the "Application Kit™."

Use "application programmer," not "applications programmer."

The application that manages your workspace and is part of NextStep® is called the "Workspace Manager™," not "Workspace."

Adobe requires that neither "PostScript®" nor "Display PostScript®" be used as a noun by itself. Follow "PostScript" with "interpreter" or "language" wherever possible, but you can also follow it with other words, as in “PostScript code.” Let the Adobe green book be your guide, since it follows Adobe’s current legal requirements. Note that Display PostScript is not a language; usually your references to it will be “the Display PostScript system.”

Differences from Macintosh Terminology

Macintosh Term	NeXT Term
active window	key window (but an application may be “active”)
to activate a window	to make a window the key window
alert box	panel
Clipboard	pasteboard
close box	close button
desktop	workspace
dialog box	panel
direction keys	arrow keys
folder	directory
inactive window	window no longer the key window (see “active”)
pointer	cursor
scroll arrow	scroll button
scroll bar	scroller
scroll box (thumb)	scroll knob
size box	resize bar
text box	text field

Technical Documentation

Reserve the word “tools” for what we call “programming tools.”

Use “header file” for files in **/usr/include**; for example: “...as defined in the header file **stdio.h**.” If the header file is in a subdirectory, give its name: “... the header file **appkit/defaults.h**.”

Avoid using event types as nouns—for example, “mouse-down” as a noun by itself, rather than “mouse-down event.”

Use “write to the standard output” and “read from the standard input” rather than “write to standard output” and “read from standard input.”

“Package” has special meaning, so avoid using it loosely.

We use the word “caret” generically, to mean a symbol indicating where something is to be inserted; the specific symbol is usually a vertical bar (|).

Use “null pointer,” not “NULL pointer.”

Call things like **pswrap** “programs” rather than “utilities.”

Use “**pswrap**-generated functions” rather than “**pswrap** functions.”

Use “rectangular coordinate system” rather than “xy coordinate system.”

Spelling and Capitalization

Use	**Don’t Use**
acknowledgment	acknowledgement
appendices	appendixes
assembly language (adj.)	assembly-language
backward	backwards
bitmap	bit map
bit-field (C language)	bit field, bitfield
bit field (Motorola processor)	bit-field, bitfield
Boolean	boolean
Businessland	BusinessLand
canceled	cancelled
canceling	cancelling
Carnegie Mellon University	Carnegie-Mellon University
CD-quality (adj.)	compact disk-quality
compact disc (or CD)	compact disk
computer music system	computer-music system
ComputerCraft	Computercraft
Control-X	Control-x (see “Keyboard and Characters” above)
coprocessor	co-processor
database	data base
double-click	double click
ensure	insure
e-mail	email
fax	FAX or Fax
file name	filename
fixed-point (adj.)	fixed point
floating-point	floating point
forward	forwards
gray	grey
hertz	Hertz
Hz	hz
highlight	hilite, highlite, or hilight
judgment	judgement
KB	Kbytes (except in Hardware publications)
400 KB (noun or adj.)	400K, 400KB, or 400-KB
kilobytes	Kilobytes
400-kilobyte (adj.)	400 kilobyte

labeled	labelled
labeling	labelling
laserdisc	laserdisk, laser disk, or laser disc
lowercase	lower case
matrices	matrixes
MB	Mbytes (except in Hardware publications)
16 MB (noun or adj.)	16M or 16-MB
megabytes	Megabytes
16-megabyte (adj.)	16 megabyte (see remark at end of table)
MHz	Mhz
MIPS	mips
modeled	modelled
modeling	modelling
multiple-click	multiple click
multiprocessing	multi-processing (but the NeXT Computer isn't)
multitasking	multi-tasking
NeXT Computer	NeXT computer
NextStep	NeXTStep (likewise for similar names)
off-line	offline
off-screen	offscreen
on-line	online
on-screen	onscreen
OK	okay
pathname	path name
PostScript	Postscript
RS-422	RS422
run time (n.), run-time (adj.)	runtime (technical documentation only)
setup (adj. or n.)	set-up
signaled	signalled
signaling	signalling
standalone	stand-alone
startup (adj. or n.)	start-up
superuser	super-user or super user
TeleBusiness	Telebusiness
time stamp	timestamp
toward	towards
two's complement	twos complement or 2's complement
upper or lower case	uppercase or lowercase
uppercase	upper case (in user documentation, use "capital")
usable	useable
user name	login name
white-space characters	whitespace (technical documentation only)
word processing application	word-processing application
writable	writeable
x-axis	x axis
y-axis	y axis
zeros	zeroes
ZIP code	Zip code or zip code
100-by-100 square	100 x 100 square

NeXT's official product names do not hyphenate between numbers and "Megabyte," as in "40 Megabyte Accelerator Drive." Every occurrence other than official product names should, however, be hyphenated. See the "Punctuation" section for more on hyphenation.

Avoid using words beginning with "non" that aren't in the dictionary, but if you do: Usually there should be no hyphen after "non."

Spell out "one" through "ten" in, for example, "four names," "six bytes," or "eight-bit field," but not in, for example, "bit 4" or "a field containing 2" (the value).

Don't spell out the value 0 but spell out "zero-length" and "nonzero."

Don't use the abbreviations "i.e." and "e.g."; spell out or avoid.

Typeface Conventions

The fonts used in our standard book design are discussed at length in the Book Design document. This section presents conventions for the use of bold and italic that apply regardless of which font is used. (A general rule is that font families—such as Times® and Helvetica®—should not be mixed within a paragraph.) There's also a subsection that gives special conventions for showing syntax.

Use italic for emphasis; for example, "do *not* call us," rather than "do NOT call us."

Put the following in bold unless they appear in an example or are listed in a table (in other words, unless they're in a context where they don't need to be distinguished from surrounding text).

- Symbols, as in the sentence "Precede the comment with **%%**."
- User names, such as **hcasabon** or **root**.
- Mail aliases and addresses.
- File names, directory names, and pathnames. Avoid long pathnames; where necessary to avoid strange line breaks, try putting them on a separate, indented line, in which case they do not have to be bold.
- PostScript operators.
- Keywords ("reserved words") in C, or whatever the language (including data types like **int** and **char**). These should be bold even when used as nouns, as in "the **id**s and **int**s"—but don't bold the "s."
- Program names and command names in UNIX (this does not include application

names).

- Variable names except NeXT-defined global variables, such as NXApp and NXDrawingStatus.

- Function names and names of macros that mimic functions. Also put the "()" following these names in bold (see below).

- Method names and the colon following them.

As shown in the Book Design document, text that the user types is often set off on a separate, indented line; however, there are many cases where what's typed is so short that it would be better to show it in-line. In that case, make the text bold. Usually it's clear in context that the user doesn't have to actually type the text in bold. To avoid confusion, when bold or italic characters do have to be typed, the text should be set off on a separate line, or enclosed in quotes. In general, we avoid quotes around text to be typed because it can look very "busy" (and may imply that the quotes need to be typed).

Do not put the following in bold:

- Menu names, command names, and control names (such as names of buttons and text fields—but see also "Punctuation" below).

- Application names, except when used as file names, as in:

  To start up WriteNow, double-click **WriteNow** in the second browser column.

- Data type names other than the standard C types; for example, NXRect and NXCoord would not be bold.

- Class names.

- Constant names.

- Extensions to file names in sentences like:

  These files have a ".m" extension.

- Prefixes, as in:

  These constants all begin with "SND_".

- References to an exact sequence of characters in an example:

  Notice that after "SND_Foo:" in this example, ...

If any of the above begins with a lowercase letter, don't capitalize it at the beginning of a sentence.

Follow each function name with **()**. Do the same for macros that mimic functions. To avoid the misunderstanding that that stands for calling the function with no arguments, don't show, for example, "call **NWFunc(**1234**)**" in the text; say instead, "call **NWFunc()** with the argument 1234."

Showing Syntax

When showing syntax and discussing what you've shown, put what doesn't vary (what should be taken literally) in bold and what varies in italic.

Data types in syntax (as in a function synopsis) are not quite taken literally (that is, the user doesn't type them) and so should be plain. (This has the benefit of making the function name stand out boldly by itself.)

For the varying, italic parts, observe the following conventions:

- Begin the term with lowercase unless there's a programming style convention for using initial uppercase. For example, class names are conventionally in initial uppercase, so a syntax term representing a class name would likewise begin with uppercase.

- Avoid multi-word terms, but if they can't be avoided, begin the second word with uppercase—unless there's a programming style convention to do otherwise. For example, in the PostScript language, words are concatenated in all lowercase, so we do the same in our syntax terms. Also, there's a UNIX convention to use all lowercase for syntax terms, so we do this in our function descriptions.

When showing the syntax of a function, type a nonbreaking space (Alternate-space) between every data type and variable name.

If you use vertical bar to mean "or" in the technical documentation, say so right there, because in many places in the documentation it should taken literally.

Punctuation

Spaces and Commas

Follow every period at the end of a sentence (and every colon) with two spaces. There's a trick for ensuring the number of spaces after a period:

1. Eliminate any three-period cases: Replace all periods followed by three spaces with a period followed by two spaces.

2. Replace all periods followed by a space with a period followed by two spaces. This

turns one space after a period into two, and two into three (to be fixed in step 3).

3. Replace all periods followed by three spaces with a period followed by two spaces.

Note: Two spaces after a period is less important in justified text, where the spacing between words varies; however, we don't justify.

Put a space after the comma between two coordinates—for example, (x, y).

In an address, put two spaces before a ZIP code.

When listing three items, such as "a, b, and c", put a comma before the " and." Likewise for "or."

Quotation Marks

Put punctuation inside quotes according to the traditional rules except where it would be incorrect for technical reasons.

If fields are labeled with all words capitalized (except embedded words like prepositions), don't enclose the label in quotes; otherwise, enclose it in quotes but omit the trailing colon, if any. For example, in Mail's Send window, refer to the To, Subject, and "cc" fields. You might refer to a Send to Whom field, or a "Send to whom" field, depending on how its label is capitalized. (Ditto for button labels.) Other examples:

> the Find field
> the "Replace with" field
> the Merge to Printer field
> the Save As field
> the "Smart quotes" field

You should, however, always use quotes around the label when it follows the word "labeled," as in: the field labeled "Find." But this should happen rarely.

Don't put quotes in a name following "called" or "named," as in

> a class called Window

This is a bit unconventional, but consider that in many cases what follows "called" will be capitalized, or lowercase and bold.

Except in code examples, use "smart quotes" (not ").

Hyphens

In general, hyphenate two-word combinations that modify a noun (as in this sentence). However, don't do so in cases where it would be "more distracting than helpful" (*Words into Type*, page 230). For example, we use "computer music system." Exceptions such as this will be listed under "Spelling and Capitalization" in this Guide.

Do not put a hyphen after the adverb in a construct like

<adverb> <adjective> <noun>

unless the adverb could be read as an adjective modifying the noun (for example, "newly created file" but "much-improved performance").

Other Punctuation

Use "—" (em dash, typed with Alternate-Shift-hyphen), not "--" (two hyphens).

Punctuate telephone numbers as, for example, (212)776-5789, or 1-800-848-NeXT.

Bulleted and Numbered Lists

Try to precede each list of bulleted or numbered items with a sentence ending with a colon, or at least say something about them being listed below. Don't just have the list appear there without warning; give it a context.

Number the items only when the order matters.

Avoid having more than one paragraph per item.

Skip a line between the items if any one of them is longer than one line.

Capitalization and Punctuation in Lists

Begin each item in the list with an uppercase letter (unless, for technical reasons, it must begin with lowercase).

Put a period, not a right parenthesis, after the number in a numbered item.

If all or part of any item is a complete sentence, end each one with a period. For example:

In this chapter, you will:

- Create a file.
- Open a document.

These are considered sentences, even though they actually complete the sentence that leads into the list. There are some who feel that long items containing clauses separated by commas ought to have periods at the end even if they're not sentences, but this doesn't match *Words Into Type* (see its list on page 38) or *The Chicago Manual of Style* (see paragraph 5.9).

Trademark Issues

This section lists NeXT's trademarks, and explains how to use trademark symbols and how to write a trademark legend.

NeXT Trademarks

NeXT™
the NeXT logo™
Application Kit™
Digital Librarian™ *(Digital Library is not a trademark.)*
Digital Webster™
Interface Builder™
Lip Service™
Music Kit™
NextBus™
NextBus Interface Chip™
Sound Kit™
Workspace Manager™
NextStep®

Using Trademark Symbols in Text

Put an unregistered (™) or a registered (®) trademark symbol after the trademark's first occurrence; however, if the trademark is the same as its owner's name (for example, "NeXT"), include the trademark symbol only when it's part of a brand (or product) name (as in "the NeXT™ Computer," but not in "NeXT, Inc."). A simple test is to insert the word "brand" after the name; if the insertion sounds right, then the name is a trademark and the appropriate formalities should be followed. If the name "NeXT" represents the company NeXT, Inc., as in "NeXT Authorized Service Center" or "NeXT customer," it's not a trademark.

Insert the trademark symbol only at the first mention of the name, even if it's on the cover page, but don't put trademark symbols in the table of contents. Similarly, if the first occurrence is in a heading rather than the body, defer the trademark symbol until the first occurrence in the body.

If punctuation follows a trademark, put the punctuation after the symbol and not before it—for example, "NeXT™," not "NeXT.™"

The following table shows the keys you need to type to generate the trademark symbols, depending on whether you're using a serifed font (like Times) or a "sans serif" font (like Helvetica). Trademark symbols are usually in a smaller point size and superscripted a bit from the regular text.

Font You're Using	**Unregistered TM**	**Registered TM**
Sans serif (like Helvetica)	Alternate-t (™)	Alternate-R (®)
Serifed (like Times)	Alternate-T (™)	Alternate-r (®)

Note: If you're using the NeXT Publications Book Design format for your document, trademark symbols on the cover page should be in 10-point and superscripted nine points. And if you use 11-point Times for general text, you can copy and paste entries from the trademark list into your text (the symbols are 9-point and superscripted two points).

Writing a Trademark Legend

In the trademark legend in any of NeXT's documentation, put NeXT's trademarks first, starting with "NeXT" and "the NeXT logo"; then alphabetize others' trademarks by owner. Here's a sample trademark legend:

Headings

There should be no more than four levels of headings within a chapter (not counting the chapter title).

Don't have sections named only "Introduction" or "Overview." Qualify the title, as in "Introduction to Drawing" or "Sound Overview."

Avoid having a heading with no text between it and the next subheading. (This is not a strict rule.)

Glossary

Italicize terms where they're defined in the text.

Alphabetize the glossary (and the index) word-by-word, not letter-by-letter. Ignore slashes. Treat hyphens as spaces. Put numbers before letters, in numerical order.

Most entries will have the following form (the examples here illustrate our standard book design for glossary entries):

> ***term***
> Partial sentence.

These forms are also OK:

> ***term***
> Partial sentence; partial sentence.

> ***term***
> Partial sentence. Full sentence.

Use this form only when necessary:

> ***term***
> Full sentence.

Don't do this:

> ***term***
> Partial sentence; full sentence.

If there's a "See also", add it as follows:

> ***term***
> Definition. See also *other term.*

Some chapters in our technical documentation are explicitly directed at programmers with a great deal of prior knowledge in a specific field (the "specific audience"). Nevertheless, programmers without the requisite background (the "general audience") may be curious enough to want to read the material, even though they may never be able to actually put it to use. In such dual-audience chapters, handle glossary entries in the following manner:

- Terms that require defining for the specific audience get italicized, defined in the chapter's text, and entered in the glossary.

- Terms that the specific audience should be familiar with, but that the general audience may well not, also get entered in the glossary, but do not get italicized or defined in the chapter's text.

- Because the first few introductory paragraphs of a chapter should be accessible to everyone, important terms in those paragraphs get italicized, defined in text, and entered in the glossary, even if the specific audience should already be familiar with them. (This is doubly important, because a chapter typically doesn't specify its audience until near the end of the introductory paragraphs.)

Cross-References

Use "see" rather than "refer to," and capitalize "Chapter," as in

> See Chapter 4.

When referring to another chapter (in the same volume), follow the chapter number by the chapter title in quotes the first time the reference appears in a chapter; after that, just refer to the chapter number. For example, the first reference to another chapter would be:

> For more information, see Chapter 3, "The NeXT User Interface."

Later references to Chapter 3 in the same chapter would be just something like "See Chapter 3."

Do the same when referring to appendices.

When referring to another chapter in a different volume of the same manual, give the volume number and chapter number but not the chapter title, even if it's the first reference. For example:

> See Volume 2, Chapter 4 for more information.

> See the description of this function in Volume 2, Chapter 4.

> See "Scheduling" in Volume 1, Chapter 8.

Don't ever refer to a specific page, chapter number, chapter title, or section title in another manual unless you know that, beyond any doubt, the manuals will always be updated and reprinted simultaneously. (Even then, it's not a good idea.) Instead, word the reference so that the reader can find it easily in the other manual, preferably in the index.

Specific References

When referring to something in another chapter, the level to which you go in the heading hierarchy should depend on how specific you feel the reference needs to be, and whether you feel the reader should first look over higher-level background material. (Many references don't have to be real specific.) For example, if you need to refer to the exact imaging conventions discussed in another chapter, it ought to be as specific a reference as possible; the interested reader shouldn't have to rummage around for the rules. If you're referring the reader to where to get more information about drawing, then your reference need only go down to the first-level heading "Drawing." (A vague reference to an entire chapter, as in "See Chapter 3 for important general information about windows," is also OK, especially if that chapter has its own table of contents, as in our technical documentation.)

Having more than two headings in a reference will be a bit awkward and would ideally be avoided. For example, these seem OK:

> See "Imaging" in the "Drawing" section of this chapter.

> See "Imaging" in the "Drawing" section of Chapter 3.

But if it's the first reference to Chapter 3, avoid this:

> See "Imaging" in the "Drawing" section of Chapter 3, "Using NeXT Windows."

You can instead finesse it with:

> See the discussion of imaging in the “Drawing” section of Chapter 3, “Using NeXT Windows.”

The same applies to any references to third- or fourth-level headings in the same chapter. But again, do this only when a specific reference is actually needed.

When referring to another section in the same chapter, say “above,” “below,” “earlier in this chapter,” or some such clue as to its location in the chapter. But if you refer to the same section later in the chapter, you can drop this.

Hardware Conventions

This section describes conventions that primarily apply to our technical hardware documentation (written by the Hardware Publications).

Terminology and Usage

Pick one name for your product. If you wish, make it into an acronym. For example:

> This manual describes the NextBus Interface Chip (NBIC).

After you define the acronym, use it consistently. Also, unless necessary for clarity, don’t be redundant; for example, don’t call it the NBIC chip.

Use “tristate,” not “three-state.”

Use “assert,” not “enable.”

Use “deassert,” not “disassert” or “unassert.”

Use “drives high/low,” not “sends high/low.”

Use “open drain,” not “open collector,” with CMOS.

Use mostly “signal,” not “line.”

Spelling and Capitalization

Signals are shown as RQST*, not !RQST or /RQST.

Bits are shown as RQST[1:0]*.

Don’t unnecessarily capitalize nouns. For example:

> Use “read word operation,” not “Read operation,” “Read Word operation,” or “Read Word Operation.”
>
> Use “start cycle,” not “Start cycle” or “Start Cycle.”
>
> Use “acknowledge cycle,” not “Acknowledge Cycle” or “ACK cycle.”

Use “DRAM,” not “Dram” or “dram.”

Temporary Notes in Drafts

A good way to indicate temporary remarks in your drafts is to enclose them in double angle brackets and, to make them stand out even more, italicize them *<<like this>>*. They’re useful as notes to yourself or to reviewers of the documentation.

Note: In Hardware publications, temporary remarks have the format (**This is a remark.)

Our convention is to use a single angle bracket to enclose things in an interim draft that will or may change in the final draft, as in:

> There are <10> classes in the Application Kit.

Similarly, we use single angle brackets to indicate where art should eventually be inserted (that is, what they enclose may or will be replaced by something else).

Tips for Writers

During the four years that I managed NeXT’s Publications group, I compiled a list of tips for writers in the group. Although the wording here may not be appropriate for all readers of this guide, you may find something useful in these guidelines on good writing habits and practices:

- Strive to be meticulous. Attention to detail is the number-one rule. All other rules derive from this. Don’t just read your draft on-line; *always* read it in print before you turn it in for review, preferably more than once. Sit quietly somewhere with it.

You'll catch more problems this way. Scrutinize it; check it against all the guidelines and any previous review comments. Read it slowly. Say the words in your mind; try to see whether there are other interpretations besides the ones you intend.

- Assume the job isn't easy. There's always a temptation to play "fast and loose" with writing; unlike programming, there are no strict rules you must obey so that a compiler will accept your program. But there's not as much freedom as you might think. There's a hidden structure, and rules that are actually more numerous and subtle than for programming languages.

- Pay strict attention to the order of things. Don't use any terms before they're explained. Know how what you're currently writing fits into everything else you've written or are going to write on the subject. This may require a lot of planning before writing, but it will save a lot of rewriting later. Decide on an order and write with that order strongly in mind. Don't leave any steps out of the logical progression toward understanding.

- Be fanatically consistent. If you're writing something that's similar to anything you've written before, or that you think anyone else at NeXT may have written before, compare. Use the language consistently. Even if you haven't read the conventions in the Style Guide lately, it should be obvious that something is wrong if you've written, for example, both "press a key" and "press a character." You'll need to pay attention and have a good memory; that's why a quiet place without any disturbances or physical discomfort is important.

- Define terms very carefully (whether or not they're formal glossary terms). If there's a glossary, work on it along with the rest of your documentation. This will force you to consider the essence of the term and to use it consistently. Every time you use a term, be sure it exactly matches its strict definition. Think of it as defining a variable in a program: You should be able to plug in the "value" (definition) of the term everywhere it's used and have it still make sense.

- Strive to be 100%, not 99%, accurate. It's true that many people may not notice a slight inaccuracy, but careful readers who will notice it will be confused. If you disagree with a review comment regarding factual content, take the time to double-check it rather than risk an error; if you still think you're right but the comment was a strong one from a reliable reviewer, follow up on it with the reviewer.

- Be aware of the flow from one section to the next, one paragraph to the next, and even one line to the next. Non sequiturs are fatal. It doesn't matter if the connection will be clear just a little later on; if connections aren't clear all along, the reader will become disoriented and no longer pay attention. The first paragraph under a heading should have an obvious connection to the heading; it's confusing when the first paragraph presents introductory material that defers bringing up the heading subject until the second paragraph or later.

- Don't substitute specifics where general concepts are needed. General concepts are harder for you to figure out and clearly state, but they will help the reader pull things together.

- Don't be intimidated by material that you deem "too technical." It's no harder to understand than lots of other things you already know. It's more an attitude problem, like math phobia. You are smart enough to understand every last detail of it, with concentration and persistence.

- If you're having a problem figuring out how to write something, you can ask me for help (before you give me a draft). If you have questions about something we discussed when going over one of your drafts, or have trouble applying any of my suggestions, discuss it with me. Don't follow a suggestion that doesn't seem to be working; you may have misunderstood or need help, or I may have inadvertently misguided you.

- Keep it clean. Leave out the obvious. Be conversational but not chatty. Avoid words that are more formal than you would use if you were telling someone the information verbally.

- Learn to think abstractly about things with great thoroughness, rather than wait until you can get hands-on experience. You won't have enough time to document everything, or to contribute to its design, if you wait that long. If you find you can't get a handle on it, draw pictures, imagine specific scenarios, break it down logically, write up notes or an outline, talk to people about it, spend far more time digesting it than you'd really like to, then do it again as soon as you feel it slipping away—don't give up.

- Consider your source material to be the Bible. If there's anything at all in it that doesn't make any sense, don't let it go by. Behind the most offhanded remark there may be a wealth of valuable information. Know when you need to ask questions rather than make judgments on your own.

- Incorporate every review comment that you reasonably can. Even if only 1 out of 10 reviewers noted it, ignoring the comment may mean that 10% of your final readers will have the same comment. Try to make the documentation work for everyone.

- Learn where you can compromise to get a draft out faster. Don't compromise on any of the above, but figure out what you can give up to get things out fast and refine them later if you have time. Prioritize. On the other hand, although fast writing is important, take the time you need to do it right. Don't rush something just to meet a milestone; admit the milestone wasn't met, and try to set the next ones more accurately based on your experience.

- Hand in something you would be proud to sign your name to as is. Imagine you don't even have an editor. The exercise in paying attention to detail is a very valuable one that will encourage good writing habits overall.