# Chapter 9: Technical Editing

## Objectives

Upon completion of this chapter, readers will be able to:

1. Differentiate between levels of technical editing—including correctness, visual readability, and content structure—and apply appropriate strategies based on editing goals and project scope.
2. Implement professional editing procedures for both hard copy and soft copy formats, including the use of markup symbols, change-tracking tools, and structural formatting.
3. Write effective, audience-sensitive editorial comments using rhetorical strategies that support collaboration with authors and subject-matter experts.
4. Evaluate and edit visual and multimedia elements for clarity, appropriateness, accessibility, and ethical presentation within technical documents.

## Technical Editing

You may find that technical editing is very different from what you expect. When people hear the word "edit," they think of rewriting an author's words; working with authors on issues such as character plot, and storyline; suggesting the most appropriate word in order to make a manuscript "sing." That's not technical editing.

Instead, technical editing is a highly rhetorical, detail-oriented process of ensuring that specialized information appears so that it is appropriate for end users, and technical editors make informed, thoughtful suggestions for improvement toward that purpose.

Technical editing is a collaborative process with authors, who are often subject-matter experts (SMEs, pronounced "smees"), to check correctness of such things as chemical formulas, specialized terminology, equations, and matchups between textual and visual elements, as well as more traditional aspects of writing.

Technical editing is a recursive process, not a one-and-done routine. Technical editors often review the same materials multiple times and have their edits reviewed before the materials are printed or posted online. Only rarely will technical editors make changes and then publish the materials immediately.

Technical editing covers a surprisingly wide variety of subjects, contexts, and materials. Job ads for technical editors seek people who can comment on—and create new—paper documents, electronic documents, images, visual designs, websites, audio and video files, and multimedia presentations, just to name a few examples.

This chapter will focus on editing text on hard copy, soft copy, and websites, but it will also provide you with concepts and techniques that you can use in graphics-heavy and multimedia editing tasks.

### General Procedure for Editing

The way you go about editing technical materials will depend on multiple factors. You will need to consider the artifact you are editing—is it mostly text? does it contain visuals? is it mostly visuals? is it paper-based or in electronic format? does it contain multimedia content? is it static or interactive?—and the type of edits that you are responsible for making. Even so, you can use the same general strategy when approaching most technical editing projects:

1. Analyze the materials' purpose, audience, format, and uses.
2. Evaluate the materials to see if they fit. In particular, consider the materials'
   * Contents: completeness; appropriateness
   * Organization: order of contents; signals about order
   * Visual design: text; lists; tables; aesthetic appearance
   * Navigability: findable, working hyperlinks; section breaks
   * Style: writing style; authorial persona; sentence structures; cultural biases; grammar; mechanics
   * Illustrations: type; construction; placement
   * Accessibility: ADA compliance
3. Set up objectives and plan your project's sequence.
4. Review the plan with the author.
5. Edit the materials.
6. Evaluate the outcome.

### Editor-Client Contracts

Sometimes, you and the technical materials' creator will work inside the same organization. In this case, your job title and job description likely already define your relationship with the creator, and both you and the creator will have set responsibilities and deadlines.

Other times, you may be editing materials for a client, a person who is not your coworker. In this case, you need to write a contract that defines your professional relationship with your client.

At the least, a contract should specify

* the type of materials you will edit
* the number of items
* the length (or size) of the materials
* the format of the materials
* the level of edit
* the deliverable (what you will return to your client)
* a schedule for completion
* your compensation

A clearly written contract benefits both yourself and your client. You will not be overworked or underpaid, and both you and your client will know what to expect and when to expect it.

As a general rule, if you are an inexperienced editor, double your estimate of how long it will take you to edit a project, and charge a per-hour or per-page rate. Once you are more experienced and know how quickly you can actually edit, you can charge a per-project flat fee.

### Levels of Edit

When you begin an editing project, avoid the temptation of diving in and making any-and-all changes that you think will be valuable. Instead, find out what "level of edit" you need to perform, and stick to it.

A "level of edit" defines how "deep" you should go with your edits. Levels range from superficial to extremely deep. Many different levels of edit can exist; experts disagree about how many levels of edit are necessary and what the different levels should involve, and some types of materials may not require specific levels of edit. Even so, you can use three basic levels for most technical editing projects:

1. **Consistency and correctness:** Edit for surface-level issues such as spelling, punctuation, grammar, word use, page numbering, cross-references, and color consistency. Changes from these edits will not deeply impact the document as a whole.
2. **Visual readability:** Edit for substantive issues such as typeface choices and consistency; graphic elements' locations, sizes, labels, and captions; and document layout. Changes from these edits may have ripple effects across a document and create new errors with consistency and correctness.
3. **Content and structure:** Edit for deep issues such as internal organization, sentence structures, logical flaws, image appropriateness, and overall meaning. Changes from these edits often require fundamental changes in the document and may create entirely new problems with other levels of edit.

When you edit any technical materials, do multiple passes through the material, moving from the deepest to the most superficial level of edit. That way, you will avoid wasting your time on marking up or correcting surface-level problems that will be deleted anyway.

If you see a problem that is outside your responsibility as an editor—for example, if you see a logical problem but you're only responsible for fixing comma splices—note the issue and contact someone with the authority to correct the problem.

### Editors' Resources

When you edit technical materials, consult a style guide or style sheet, and create a style sheet of your own.

#### Style Guides

A style guide is an existing, authoritative source that lays out rules for the materials you are editing. For example, you have almost definitely used a dictionary at some point in your life, and if you have taken a first-year composition course, you have used a writer's handbook. Both of these examples are style guides.

Many technical editors use their employers' own in-house style guides, but many technical editors also use commercially available style guides such as the *APA Publication Manual*.

Specialized style guides for highly technical subject matter also exist. If you are editing materials that require specialized knowledge, consult an appropriate style guide. For example, if you're editing documentation for factory-control equipment that will be exported to Russia, refer to The English-Russian Dictionary of Mechanical Engineering and Industrial Automation.

Always be prepared to justify your edits with a style guide reference. If you make up your own rules or follow your gut instinct instead of following a style guide, your author may reject your edits, or worse, you may introduce new errors.

#### Style Sheets

Style sheets are small-scale, local style guides that provide consistent, quick-reference answers to common problems. Technical editors often develop style sheets to cover separate-but-related projects or different phases of a major project, and to make sure that all the editors on a project are following the same rules.

You should compile your own style sheet every time you edit anything. Do not simply list every error you encounter. Instead, list recurring errors or problems with answers that you need to look up frequently, and alphabetize the contents to make them easy to navigate.

**Example #1: List Format**

**Terms**

* bell hooks: no caps
* bibliography: no italics, no quotation marks
* INTERPOL: all-caps if the police agency
* Interpol: initial cap if the band
* PowerPoint: two caps, no space between words
* smooth: no italics, no quotation marks
* transistor: no italics, no quotation marks
* teh awsum: italics, as spelled; emphasize humor
* twin-jet: hyphenate when used as a compound adjective before a noun
* vertiginous: spelling *(NOTE:**Only include spellings if you’re likely to misspell them again)*

**Numbers**

* one million: spell out large round numbers
* one thousand: spell out large round numbers
* twelve: spell out numerals from 0-100, inclusive

**Example #2: Table Format**

|  |  |
| --- | --- |
| **A-E** | * bell hooks – no caps * bibliography – no italics, no quotation marks |
| **F-J** | * INTERPOL – all-caps if the police agency * Interpol – initial cap if the band |
| **K-O** |  |
| **P-T** | * PowerPoint – two caps, no space between words * smooth – no italics, no quotation marks * transistor – no italics, no quotation marks * teh awsum – italics, as spelled; emphasize humor * twin-jet – hyphenate when used as a compound adjective before a noun |
| **U-Z** | * vertiginous – spelling |
| **Numbers** | * one million – spell out large round numbers * two thousand – spell out large round numbers * twelve – spell out numerals from 0-100, inclusive |

### Strategies for Writing Comments

When you edit technical materials, do not simply insert corrections unless the edits are simple or you have explicit permission to make final decisions. Instead, write comments to the author and suggest changes.

Before you write the comments, analyze the person you're writing to. Who is the author that created the materials you are editing? How will this person react to your comments? People are often very sensitive to criticism of their writing.

When you write the comments, actively think about the words and sentence structures that you use. Some authors are more open to criticism than others, but even receptive authors will ignore weak comments and balk at rudely stated commands.

Write your editorial comments using the strategies that Mackiewicz and Riley (2003) suggest:

| **Strategy** | **Example** | **Comment** |
| --- | --- | --- |
| Opinion | *I would use Verdana for the document's typeface.* | State your opinion if you mean the author should make a change. |
| Suggestion with an active modal verb | *You should probably use Verdana as the document's typeface. It'll make the text more readable onscreen.* | Combine a strong suggestion with "should," will," or "ought" if you mean the author should make a change. You can include a "downgrader" such as "probably" to soften the tone. You can, but don't have to, explain the payoff. |
| Command | *Use Verdana as the document's typeface, please. It'll make the text more readable onscreen.* | Issue a command if you mean the author should make a change. You can include a "downgrader" to soften the tone. You can, but don't have to, explain the payoff. |
| Possibility statement with an active verb | *You could use Verdana as the document's typeface. That's just an idea. It would make the text more readable onscreen.* | Make a suggestion with "can" or "could" if you are suggesting a non-mandatory option. You might also state the payoff. |
| Question | *Could you change the document's typeface to Verdana?* | Ask a question only if you don't know the answer. Otherwise, avoid this strategy. |
| Suggestion with a passive voice modal verb | *The document's typeface should be changed to Verdana.* | Avoid this strategy. |
| Possibility statement with a passive voice modal verb | *The document's typeface could be changed to Verdana.* | Avoid this strategy. |
| Hint | *Using a sans serif font for a document that will appear onscreen increases the document's readability.* | Avoid this strategy. |

### Strategies for Marking Up Technical Materials

When you edit technical materials, your specific actions will depend on the type of editing and the materials' format.

#### Copyediting vs. Proofreading

Technical editors help develop technical communication artifacts as well as review them just before they are published. Before you begin editing, make sure you know which approach you should take.

Editing during the developmental phase is called **copyediting**. This type of editing may involve "shaping" the document through deep edits and multiple comments to the materials' author. Documents that are being copyedited in hard copy are often (but not always) double-spaced.

Editing during the pre-production phase is called **proofreading**. Ideally, proofreading should only require a superficial level of edit because it requires an editor to look for differences between the approved "dead copy" that has been edited multiple times and the first printed proof version—the "galley"—that will be reproduced and published. Documents that are being proofread in hard copy (on paper) are almost always single-spaced.

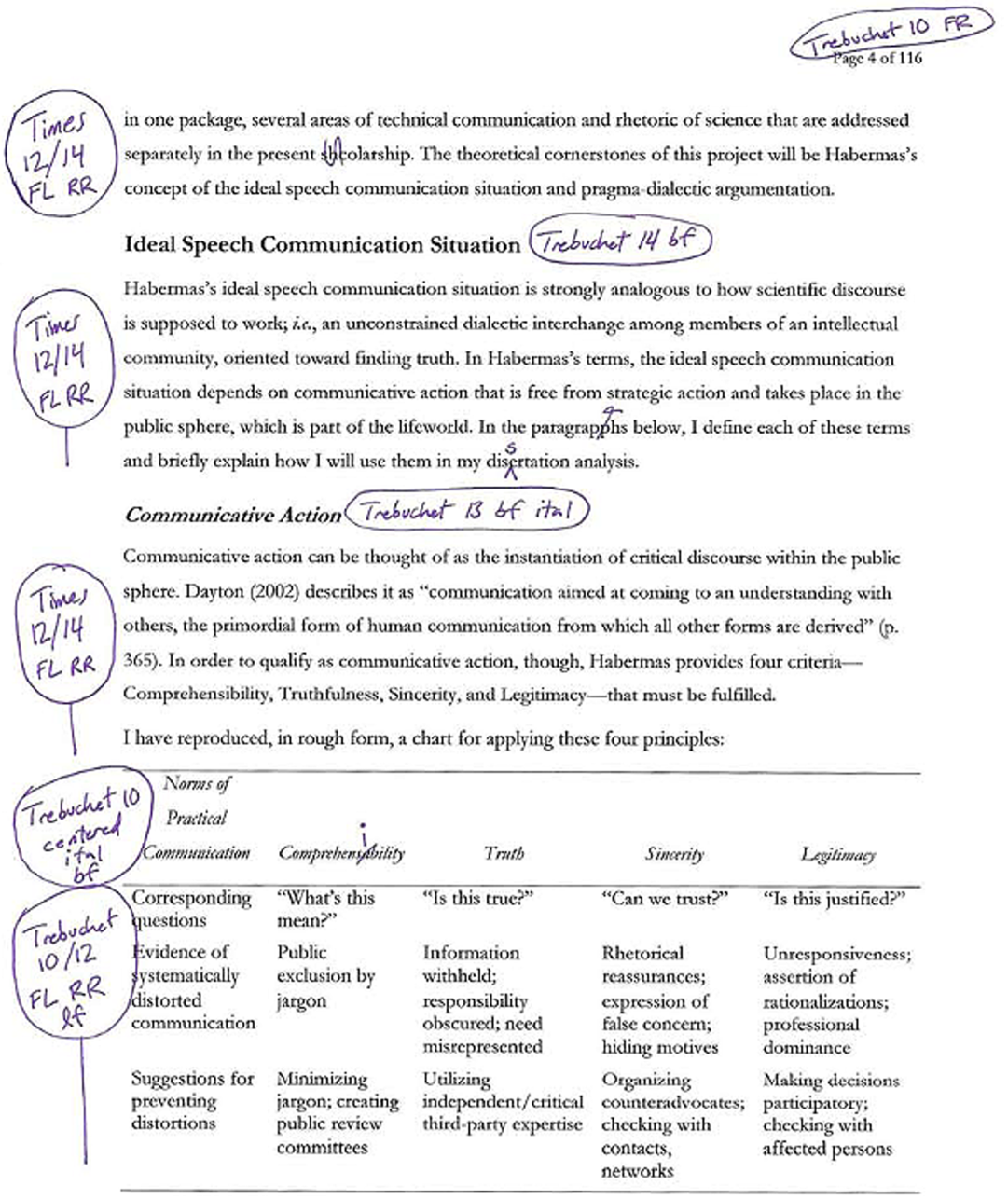
#### Procedural Markup vs. Structural Markup

Technical editors use different types of markup on text that depend on the editing goals and the edited materials' format. These approaches are complementary, not opposite.

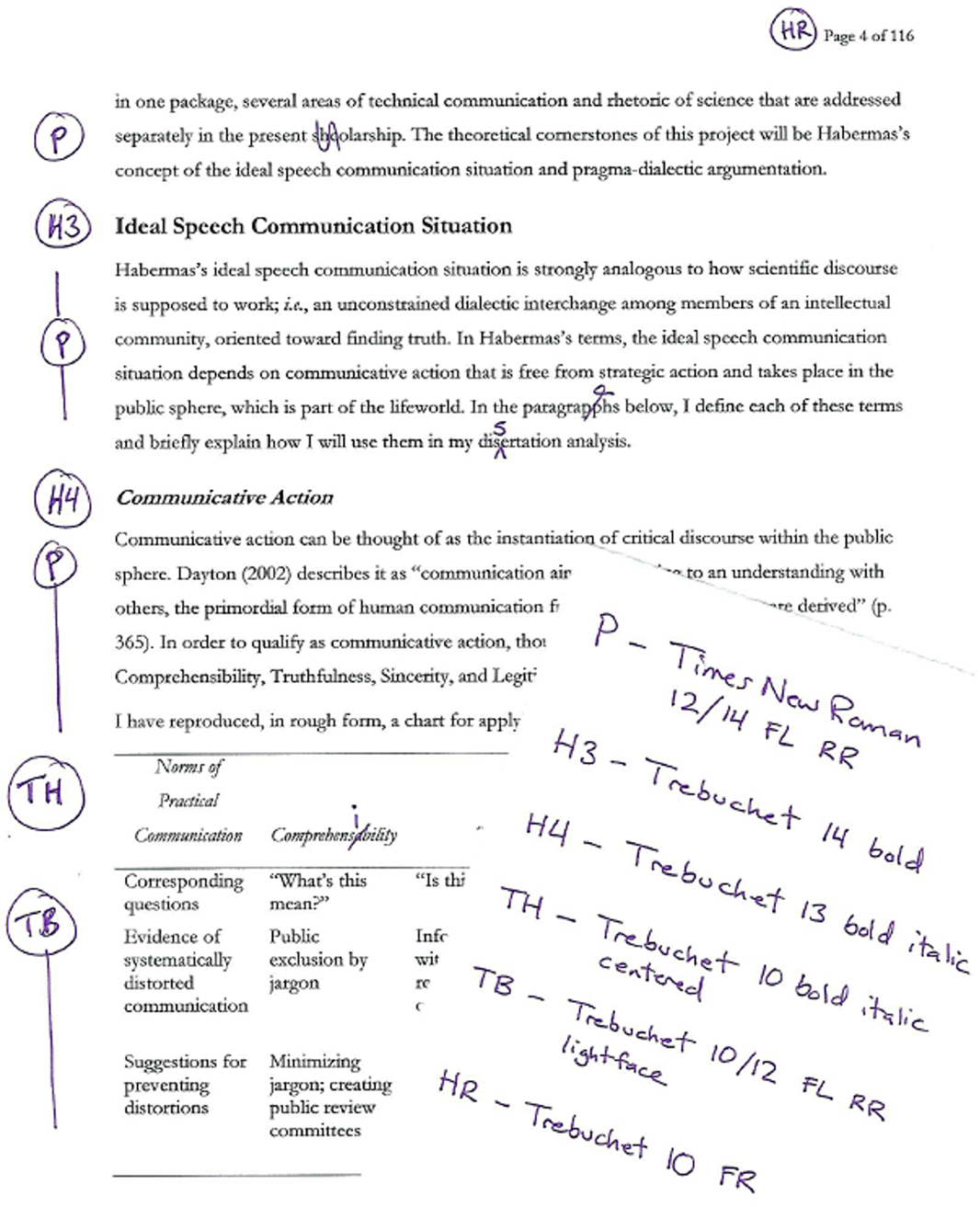
**Procedural markup** involves going through a document and marking specific changes. A common example is correcting misspelled words or deleting blank spaces. You may also use procedural markup to provide instructions for changing a document's layout and design.

**Structural markup** involves "tagging" sections of a document to indicate they belong to specific categories. It is akin to using the Styles function of MS Word.

You can also combine the two approaches by using procedural markup to indicate textual changes and structural markup to indicate formatting changes.



*Figure 1: Example of Procedural Markup*



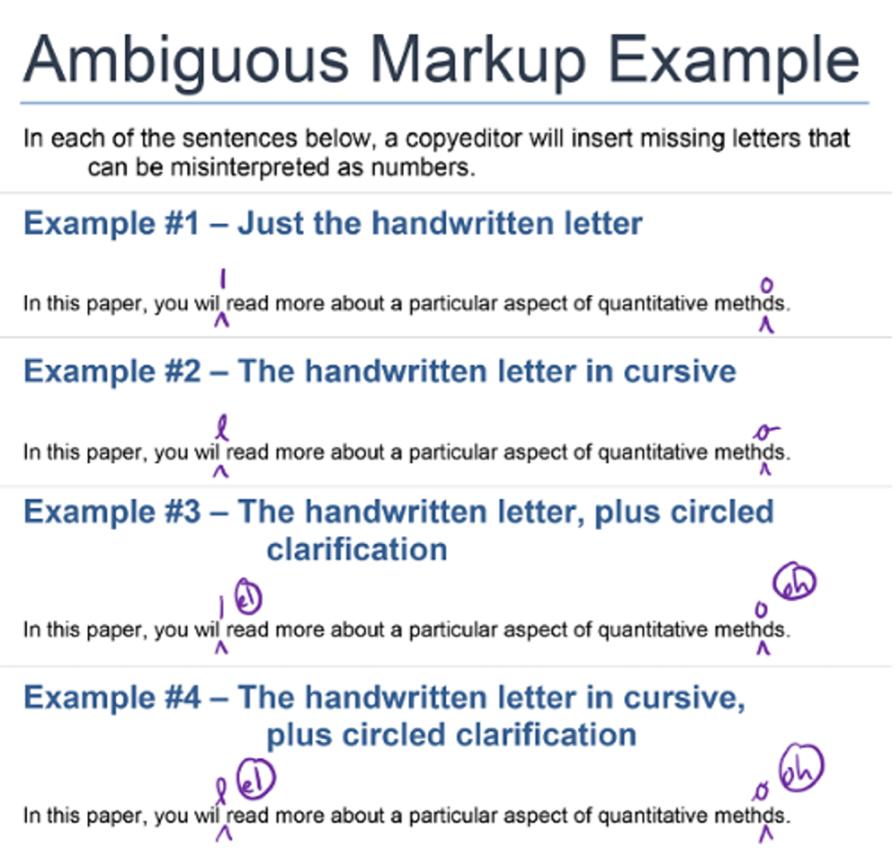
*Figure 2: Example of Structural Markup*

### Hard Copy Materials

It is becoming less and less common for editors to work in hard copy (on paper), but it still happens. You may find that editing on paper is easier on your eyes, or that until you learn how to use a program's editing tools, editing on paper may be faster.

If you do choose to print out and mark up technical materials, you should follow a few standard procedures.

* Mark changes to text inside each page's main body. (Except if you're proofreading. In that case, mark changes in the margins.)
* Mark changes to layout in the margins.
* Write comments to the author in the margins and label them as comments. "AU:" is a common label.
* Circle any marginal notes that are instructions.
* Use standard marks that other people will understand, not your own made-up marks.
* Choose the simplest markup.
* Clarify potentially ambiguous marks. (For example, if you insert a lower-case letter L, write it in cursive, and/or circle the letters "el" next to it.)



*Figure 3: Example of Ambiguous Markup*

* Be consistent and mark every instance of an error.
* Use a bright-colored (not blue or black) pen or pencil, with a medium tip.
* Erase all stray marks. They can be misinterpreted as instructions to make changes.
* Be neat. Scribbles, squiggles, and smears will only confuse the author and/or your fellow editors, and you may cover up important items.
* If you use structural markup, provide a legend that specifies each tag's formatting requirements.

### Soft Copy Materials

More and more often, technical editors work in soft copy (on a computer). Doing so lets you avoid double-handling documents, erase mistakes, revise comments and markup, track versions easily, and automate repetitive tasks.

If you edit in soft copy, you should follow slightly different standard procedures:

* Use programs' built-in commenting tools to write comments to authors.
* If you edit text on a word processor (for example, MS Word), activate the program's change-tracker and actually make changes. Don't just mark problems.
* Apply structural markup instead of just marking for it. Use document templates, high-level formatting tools (such as MS Word Styles), and/or tagging languages (such as HTML—see Chapter 22).
* Use find-and-replace tools to fix repeated errors.
* Use accept/reject functions to incorporate or reject changes and delete editors' comments.
* Toggle between viewing the edited document with markup and the document without markup. Without the change-tracker's highlighting, you may see new problems "hiding" in plain sight.

### Hard Copyediting and Proofreading Marks

Editors in many disciplines use two fairly standard sets of marks that you can use to tag hard copy documents. One set is specifically for **copyediting**; it assumes that the edited document will be double-spaced, with lots of room between the lines for an editor's scribbling. The other set is specifically for **proofreading**; it assumes that the edited document will be single-spaced.

There is some crossover between copyediting marks and proofreading marks, but they are not interchangeable. Keep them separate.

[Copyediting Symbols](bit.ly/45qXuJV)

[Proofreading Symbols](https://bit.ly/4fqIG2F)

### Special Ideas for Editing Visual Materials

Most of the concepts and techniques described in this chapter focus on editing text, but they can also apply to other technical materials. This section will address ideas specific to editing visual elements.

Consider these six concepts when you edit visuals:

#### Appropriateness

* Decide if text, an image, or a combination of the two is most effective.
* Match the type of image (e.g., table, bar chart, scatterplot, pie chart, Gantt chart, flowchart, map, line drawing, cutaway, cross-section) to the idea/thing being discussed.
* Match the illustration's emotional tone to the subject matter's emotional tone.
* Follow established design conventions.

#### Clarity

* Use contrasting colors. Bright colors and dark shades attract the most attention.
* Make sure the visual elements' layout follows consistent rules on the screen or page.
* Omit any "non-data pixels" that do not carry information.
* Place an illustration next to the text that references the illustration.
* Explain every illustration's content and relevance in the text, preferably before the illustration appears.
* Label and caption every illustration.
  + Use the correct type of label.
  + Use a sequential numbering pattern.
  + Number tables and figures separately.
* Include "white space" (blank space) around illustrations.

#### Emphasis

Use arrows, callouts, and boxes to highlight elements that a user will find important, but don't go overboard.

#### Ethics

* Avoid images that exclude categories of people or play to stereotypes.
* Avoid inhumane images that downplay effects on people.
* Follow copyright law and create, cite, and/or pay for images.
* Use undistorted images of the actual object.
* Use realistic numbering scales that display quantitative differences in context.

#### Size

* Strike a balance between making images large enough to see details but not so large as to waste space.
* Minimize images' file sizes but avoid making the images grainy.

#### Cost

Compare the cost of publishing images to your printing budget. Color pixels are free, but color ink is expensive, and projects that require a professional printer can be very expensive.

## Attribution

This chapter is revised from the first edition of *Open Technical Communication*, Chapter 8: “[Technical Editing](https://alg.manifoldapp.org/read/open-technical-communication/section/19abea6b-932a-4efd-a70d-9f1123dd4b08)” by Jonathan Arnett, which is openly available under a Creative Commons Attribution license.

## AI Assistance Notice

Some parts of this chapter were brainstormed, drafted, and/or revised in conversation with ChatGPT 4o and Google Gemini 2.5 Flash. All AI-generated content was reviewed and revised as needed by a human author.