

Concise (Public)

How to write the no-nonsense version of your document.

by Aleks Haecky (& made more concise by Katarina Stenstedt)

[Writing docs is like programming](#)

[Technical writing is like poetry](#)

[Avoid writing docs](#)

[Write docs that get the job done](#)

[Be efficient](#)

[Be concise](#)

[Be heard \(otherwise, what's the point....\)](#)

[Optimize tools and processes](#)

[Get feedback](#)

[Publish](#)

TL;DR

Technical writing is about getting the job done.

Concision respects the reader, saves time and money, and results in beauty.

Writing docs is like programming

- Every word serves a function.
- Efficiency matters.
- Instructions need to be correct and precise.
- Good architecture supports clarity and maintenance.
- Bugs are inevitable. Track and fix them.

Technical writing is like poetry

- Every word matters.
- Rhythm matters.
- Order matters.
- Formatting and presentation matter.

Avoid writing docs

- Fix your code to read like a story: redesign, refactor, rename, comment.
- Fix the UI.
- Make a diagram or short video.
- Cancel the project.

Write docs that get the job done

- Get (mostly) correct content to users.
- Unless “perfect” saves lives, “good enough” is good enough.
- Decide up front what is good enough. Usually, 90%.
- Create a one-sentence pitch for your content.

- Be inclusive and culturally sensitive when giving examples.

Be efficient

- Know how to spell. Trust the spellchecker to catch what you miss.
- Know grammar. Because grammar checkers have limits, and people care.
- Know punctuation. Punctuation marks follow the breath, not the rule book. Mostly.

Be concise

- Only use words that matter.
- Eradicate jargon and corpspeak.
- Watch for [the curse of knowledge](#).
- Edit mercilessly.
- Be specific, certain, practical, and direct.
- One topic per sentence.
- Eliminate choice in instructions.

Be heard (otherwise, what's the point....)

- Speak to your reader.
- Benefits get readers excited. Your excitement is irrelevant.
- Make your readers feel smart. Nothing is simple, easy, or quick.

Optimize tools and processes

- Use the tools that work for you.
- Minimize contributors. Give each a focused task.
- Choose one person to make all final decisions.
- Let one editor unify the content.
- Simplify your publication process.
- Only negotiate what matters to achieve the doc's purpose.

Get feedback

- QA time should be proportional to audience size and doc longevity.
- Trust your editor.
- Don't let anyone read it until you are done with the first draft.
- The purpose of feedback is to improve your document, not to make reviewers happy.

Publish

- Always attribute art, images, and diagrams. Never use if you don't know the origin.
- If you can, get graphics redone professionally.
- Use callouts and captions to make diagrams translatable.
- Use strong colors and contrast for accessibility. Prefer black on white.
- Choose titles and headings that summarize content.
- Publish where your audience can find it.
- Crowdsource maintenance.