UnTechnical Writing

How to Write About Technical Subjects and
Products So Anyone Can Understand

Also available in *the UnTechnical Press Books for Writers Series*

The User Manual Manual—

How to Research, Write, Test, Edit and
Produce a Software Manual

UnTechnical Writing

How to Write About Technical Subjects and
Products So Anyone Can Understand

By Michael Bremer

UnTechnical Writing

How to Write About Technical Subjects and
Products So Anyone Can Understand

By Michael Bremer

This book is dedicated to Linda.

It could not have existed without support and inspiration from
Barbara, Jeff, Wendy and the "other Michael."

Special thanks to Richard, Tom, Debbie, Laura, Bob
and a few others who know who they are.

Published by:

UnTechnical Press

P.O. Box 272896 • Concord, CA 94527

Contents

Foreword: The Need for Change

Technical writing, as it is generally practiced today, is a well-defined, excellent method of communicating technical information to a technical audience. The problem is that the audience has changed and expanded.

Technology is no longer just for technologists. Yes, there are a lot of engineers, scientists and technicians out there who need to read and learn more, but today, technology is all-pervasive in our society.

Look into any living room. Look at the merchandise at the local mall or department store. Technology's everywhere: camcorders, VCRs, surround-sound audio systems, alarm systems, computers, word processors, spreadsheets—technology for work, technology for entertainment.

The average citizen is becoming more and more dependent on technology. Do most people think of themselves as technologists? No. They are "normal" people who have no interest in engineering degrees, but find that they must deal with techno-stuff on a regular basis. That means absorbing a lot of technical information.

And that's where today's technology writers come in. And that's why, when we write for a nontechnical audience—that is, for most people—we need to take a different approach: the UnTechnical Writing approach.

This book introduces writers, (and aspiring writers, producers, product managers and others who are involved with explaining and selling technology) to the skills, knowledge and attitude that they need to produce clear, understandable explanations of today's technology.

Beyond the actual writing, the UnTechnical approach encourages writers to break out of the Technical Writer box, and to think and learn and do things that, in many companies, writers aren't expected to think, learn or do. Things that overlap with the duties of people in design,

marketing, customer service and business. Things that will help writers and companies better understand and communicate with the non-technical world.

> "Any sufficiently advanced technology is indistinguishable from magic."
> — Arthur C. Clarke, Technology and the Future

> "Any sufficiently advanced technology is indistinguishable from a rigged demo."
> — James Klass

Introduction

"I love being a writer. What I can't
stand is the paperwork."

— Peter De Vries

"Writing is easy; all you do is sit
staring at a blank sheet of paper
until the drops of blood form on
your forehead."

— Gene Fowler

This chapter, as a well-behaved introduction does, introduces the
subject, explains the purpose and structure of the book itself, and sends
you off on your merry way, over the pages and through the chapter.

About UnTechnical Writing

Why UnTechnical?

There's a lot of technology in today's world, and there's a lot that has to be written about it.

I've coined the term *UnTechnical writing* to refer to the writing about technology that is intended to be comfortably read and understood by the nontechnical consumer audience, as opposed to the existing term, *technical writing,* which has a pretty bad, even scary reputation outside of technology circles.

We've all—no matter how technical we are—been frustrated and angered by technical documentation at one time or another, and the less technical we are, the more it happens. The goal of UnTechnical writing, its techniques, skill set and attitudes, is to eliminate the frustration, anger and confusion the average person feels when confronted with technology and technical writing. And, along the way, lessen the frustration, anger and confusion that the writer feels when confronted with uncooperative co-workers and corporate cultures that just don't care about the writing.

UnTechnical writing techniques apply to any writing or writer that touches on technology or any complex subjects. This includes everything from user manuals to press releases to help systems to advertising. Many of the techniques apply to any type of writing.

> "For a list of all the ways technology has failed to improve the quality of life, please press three."
>
> — Alice Kahn

What If You're Not a "Writer"?

Forget your job title for now.

If any part of your job responsibility is to teach or explain anything to anyone, then, for the part of your job when you prepare and organize information, you're a writer. If you hire, contract or manage people who teach or explain things, then you're a writing manager.

If you are in any way involved with both technology and the mass market, you'll find useful information in this book.

Background—the Origin of UnTechnical Writing

I gained the UnTechnical perspective over a 10-year period working as a writer and manager in the entertainment software industry. Software manuals and their writers have been regularly maligned. Often rightly so. In fact, the first words out of my then future brother-in-law's mouth when we met were, "So, you write computer manuals. I've always wanted to kill one of you guys."

And this sentiment is not a new one:

> "...Then anyone who leaves behind him a written manual, and likewise anyone who receives it, in the belief that such writing will be clear and certain, must be exceedingly simple-minded..."
>
> — Plato

OK, back to the story.

Always interested in writing of almost any type, I cut my technical writing teeth while working as a technician/designer at a small communication electronics firm. I wrote data sheets, assembly and testing instructions and product-operation handbooks. They were written for our in-house technicians and for our customers—engineers who integrated our products into larger systems. They were short, to-the-point, by-the-book and could cure insomnia at 100 paces. But they gave the needed information to the target audience. The writing was an arduous, boring task, but it was only part of the job, so it was bearable.

After a number of years in that field, I fell into a writing job at Maxis, a start-up software company that made simulation games. In the new job, in spite of the fact that I was writing about games, I had far more technical information to convey. I felt that I needed to take another approach to my writing for three reasons:

1. The audience was different—I was writing for people who may not have engineering degrees.

2. It was an entertainment product—the customers bought the product to relax and have fun, not to use at work, further their careers or pass a college class.

3. These would be fairly long documents, and would take a lot of time and effort. If the process was going to be as boring as my previous technical writing, I'd probably kill myself. So I was determined to enjoy the writing, and write something I'd enjoy reading.

In my first few writing endeavors at this job, I did get a number of complaints. Technical writers would scold me in letters or phone calls about how the rules say this and that, and would tell me how I should have written the manual. But I got a lot more calls and letters from customers who actually *enjoyed* reading the manuals. What with all the clichés about how people never read manuals, and especially not for games, I figured that these fine people were tasteful, brilliant and generally wonderful, but few in number.

After a few years, the company had grown quite a bit. We had shipped millions of games all over the world. I had a staff of three writers working with and for me. Then we got the results of an industry survey. It found that in the software industry as a whole, only 11 percent of the people surveyed read their manuals. That sounded about right. But it also reported that 70% of our company's customers read our manuals. This was delightful, and shocking. I even had to stop billing myself as "the least-read million-selling author in the world."

I figured I was doing something right.

Over the last 10 years, I've done a lot of writing, and hired, trained and edited a number of other writers. Some of the training was by-the-book: here's how we do things around here. Some of it was explaining

the local processes. But a lot of it took the form of conversations about our typical customer, the philosophy of writing, communication, working with others in the company, responsibility to company, responsibility to readers, and how to get your job done with the least pain and stress.

This book is a condensed crash course based on those conversations.

About This Book

Who This Book Is For

- Technical writers who want to start writing for a nontechnical audience.
- Writers who already write for a nontechnical audience, and want to do a better job.
- Anyone who is new to writing and wants to write about technology.
- Anyone who has designed a product and needs to write the manual, instruction booklet, help system or even an explanation of the product to get funding.
- Anyone who manages, hires, contracts or subcontracts writers for high-tech writing.
- Anyone who needs to explain anything complex or technical to a nontechnical audience.
- Anyone who wants to understand what it's like to work and survive as a writer in the high-tech world.

What This Book Won't Do[1]

This book won't teach the basics of writing.

This book won't teach you how to type or how to use a word processor.

This book won't teach grammar.

This book won't cover writing for programmers, engineers or scientists, a.k.a., standard technical writing.

What This Book Will Do

This book will try to make clear the special circumstances of and techniques for writing about technology for the consumer audience.

This book will give you an insider's view of what it's like to work as a writer in a high-tech company.

---

[1] *There are a lot of other books that can help you learn all these things. Check the Recommended Reading section.*

"Don't judge a book by its cover, but by its coverage.[2]"
— Nobody That'll Admit It, But Definitely Not a
Graphic Artist

A Personal Note

Unlike technical writing—but very UnTechnical in nature—this is a personal book. In it I tell of personal experiences, battles won and lost, lessons learned and my own philosophy of writing and the writing life.

There is no absolutely right or wrong way to write. What I state here are the things that I believe, and that have worked for me on one or more occasions. I approached this book as if I were summarizing all the information I wanted to share with a new writer on my staff, and I offer you the same challenge that I offered them: unless you can come up with something better, try it this way. If you can come up with something better, teach me.

Feel free to contact me about this book and about your writing experiences, tips, tricks and methods. Contact information can be found later in this book.

Coming Soon to a Page Near You

The next eight chapters of this book will help you write, work, organize, present, wrangle, philosophize, craft, finish and otherwise deal with the work, people, places and problems related to your writing.

If you have basic writing skills, have a love of things technical, and enjoy explaining things to people, the information in this book will help you produce better work and enjoy your writing. You may even get some fan mail.

The information I present is grouped into these chapters:

- **The UnTechnical Writer** covers writer selection, skills, development and attitude.

- **The Nontechnical Reader** covers ways to identify and think about your target audience.

- **UnTechnical Writing** presents a number of handy hints, tips, tools,

---

[2] *Which brings to mind the time the Supreme Court justice was arrested by mistake while participating in an undercover sting. He was, of course, released after his lawyer reminded the court that you "Don't book a judge by its cover."*

techniques and suggestions that can save you time and pain while making your work more effective.

- **Editing** covers the ways and means of fine-tuning your writing and confirming its accuracy.

- **People and Politics** helps you handle the other human beings that you have to deal with to get your work done.

- **Layout and Graphic Arts** covers the polishing and presentation of the document, as well as dealing with graphic artists.

- **Interface Design** covers the basics of understanding and designing products, hardware and software, so they are better suited to humans—even and especially nontechnical humans.

- **Philosophy** is where I really pontificate on lessons learned over many years of writing.

- **Exhibits** are sample forms, lists and worksheets that may be helpful to you in your writing.

A Warning

A lot of this book encourages you to break out of the Technical Writer box, and think and learn and do things that, in many companies, writers aren't expected to do.

This can be liberating, but it can also cause problems. Take it slowly. You can't declare yourself a new person and expect a company to instantly change their ways of doing and thinking. Change is a slow process.

Don't lose your day job while working towards your goals.

The UnTechnical Writer

"Why do writers write? Because it isn't there."

— Thomas Berger

"I'm writing a book. I've got the page numbers done."

— Steven Wright

This chapter talks about the person who does the writing (whether or not the word writer is in their job title), the skills they need and how to get them.

Roles, Goals and Attitude

Who Should Write About Technology and Technical Products?

The obvious answer is: a writer. But in the real world, who actually does the writing?

The Draftee

Often, product documentation for both hardware and software is written by whoever is around and appears to the powers that be to have the time. If there is no staff or team writer, then this could be the producer, programmer, designer, engineer, artist or almost anyone else.

This can work, as long as the person stuck with the job:

1. communicates well in writing,

2. has and takes the time to do a thorough, well-tested, well-edited job, and

3. relates to the audience.

Relating to the audience is a balance between too much technical detail and too little fundamental information.

Extremely technical experts on the product, programmers, for instance, like to describe the intricate internal details that they love and spend their lives with. This is great if the audience is another programmer or someone whose job it will be to maintain the product, but confusing and frustrating for the typical consumer who buys the VCR, word processor or game.

At the other end of the scale, anyone who has been working on a product for months or years easily forgets what it's like to approach the product as a total unknown. After living with it for so long, the basic aspects of the product seem so obvious that they're hardly worth men-

tioning. The person buying the product and opening it for the first time would disagree.

It's possible to make both these mistakes at the same time. I've read web pages and press releases announcing new products that list the new features and the patented technology that makes the product so great, but nowhere state what the product does.

"Official" Writers

Sometimes a contract writer is brought in at the last minute and given very little time to learn the product through and through, figure out what the user needs and deserves to know, organize the information, write it, test it, rewrite it, guide it through layout (or do the layout), and do a good job. As frustrating and difficult as this is, it's part of the gig. The contract writer who can come in, grasp the gestalt of the product and get the job done on a tight schedule is the contract writer that gets a lot of jobs.

Under ideal circumstances, the person charged with the writing, whether on staff or contract, will be at least somewhat involved in the project for the last $1/4$ of development time, if not all along the way. This gives the writer a good grounding in the product ahead of time, so, in that last rush of development crunch-time, the writer can concentrate on writing and accuracy, and not have to figure out the basics. It will also allow the writer to suggest possible interface and design changes that will not only make the product easier to use, but may also eliminate many pages from the manual, which saves both time and money.

Writer As Communicator

This may seem a sacrilege to many writers, but I believe that the world would be a far better place if there were less technical and UnTechnical writing. Don't worry, there will always be a need for writing and writers, but I hope that a lot of what we do now becomes unnecessary.

If products are really well-designed with intuitive, self-explanatory interfaces, they won't require manuals, or at least not manuals that have extensive sections explaining the basics that should be obvious in a half-decent product.

The classic case that illustrates this point is the door. You don't notice well-designed doors. You reach for the handle or knob or whatever,

and your hand knows what to do—push, pull, twist, shout, etc.—and you're through. But a badly designed door will give you the wrong message. You'll pull when you should push, or vice versa. And that's where the unnecessary writing comes in. Some doors actually need an instruction manual. The manual may be only one word—push or pull—but it shouldn't be necessary. If the door hardware is designed well, you won't have to stop and read instructions before using it. The presence of that word on the door advertises the fact that the door hardware wasn't tested with actual door users. And that's typical of the many products, hardware and software, that require long, tedious manuals to make up for the fact that the product wasn't designed well or tested with the intended end user.

And who's going to change the way of the world? You are. Yes, you, the current and future technical and UnTechnical writers of the world. You've got the communications skills and the technical background. All you need is some knowledge, experience, a lot of patience, and the right mindset. The knowledge comes from studying product and interface design. The experience comes from working with others from whom you can learn both the right and wrong ways to do things. The patience comes … if you wait for it. The mindset is knowing that you are more than a button-explainer—you are a communicator.

Even now, if you think of yourself as a communicator who passes on information instead of a writer who puts words on paper, you may be able to insert bits of necessary information into the product itself instead of into the manual.

The UnTechnical Writer's Goal

Your goal as an UnTechnical writer is to communicate. To get information from point A (those that know) to point B (those that need or want to know) as quickly, easily, completely, painlessly and even enjoyably as possible.

That's it.

Don't worry about impressing your professor.

Don't worry about winning awards.

Don't worry about creating art.

Just get that information into the customer's brain, ready to use, right away.

The information you pass on can be in writing or pictures, in manuals, on labels, on reference cards or in the design of the product itself. Don't limit yourself. Communicate.

> "The virtue of books is to be readable."
>
> — Emerson

Point of View: The Writer's Roles

In fiction, the writer writes to entertain or reveal something about life from any point of view imaginable, from child to dead spirit to deity to alien. In standard technical writing, the writer writes to inform and pass on mastery of a system or subject from the point of view of one technical professional to another.

In UnTechnical writing, the goal is also to inform and pass on mastery of a product, system or subject. There are a number of points of view that work, and some that definitely don't.

When writing for the consumer audience (the whole point of this book), avoid the standard technical-writing point of view. Since the target audience isn't a technical professional, it is easy to make assumptions about their knowledge base and leave out important basic facts, put in too many details or move too fast. You should also avoid writing from the point of view of the Master deeming to pass on knowledge to the unwashed masses (unless you can do it with enough self-deprecating humor).

Choosing your point of view depends on the project, your mood and your personality. Unlike in fiction, where point of view needs consistency and deeply flavors the whole project, in this type of writing, point of view is more of a writer's internal state of mind. It's closer to role-playing than the standard literary point of view, although the role you pick doesn't need to be—and probably shouldn't be—obvious to readers.[3] Point of view is for you; it's a tool for setting and controlling the document's level of technology and rate of delivery.

---

[3] *For the right product and audience, you may want to make your internal role obvious. For instance, present the document as a travel guide to visit the land of [insert your product name here]. This kind of thing works better with younger audiences. If you do try it with adults, don't be too heavy-handed, or make them wade through too much fluff. It gets old fast.*

As you think about how you'll tackle a project, try on these roles for size, and see what fits you and the project:

- **Writer as translator**—you're the link between a world that speaks Technese and people that don't know the lingo. Your job is to translate technical words, concepts and processes into normal language.

- **Writer as host**—you welcome your guests to a new, scary, complicated place. Make them feel at home. Treat them with respect. Talk to them in words that they understand and feel comfortable with. Not only the words, but also the format, feel and media as well.

- **Writer as friend**—you're trying to make your friends feel comfortable while helping them master this new experience.

- **Writer as nerd next door**—you're the guy who knows and loves technology, and you're the one everyone on the block comes to when they have technical questions. You know the deeper inner workings of the technical universe, but know most of your neighbors don't care about that, so you reassure them, then tell them what button to push to get the result they want (tutorial) and give them more information if and when they ask (reference).[4]

- **Writer as teacher**—not the cranky teachers you hated, but your favorite teacher that lit that spark in you and made you want to learn.

- **Writer as intrepid explorer**—you've been down this dangerous road before, and are documenting your discovery process for the readers, allowing them to follow and see the sights, yet avoid the pitfalls. This is especially useful for explanations of very complex systems.

- **Writer as tour guide**—you're guiding a group of tourists through a dangerous, foreign land. You've got to watch out for them, even protect them, while showing them a good time.

---

[4] *All those billion-dollar computer companies should set aside some sort of retirement fund for Nerds Next Door. Judging from the complexity of early computers and the incomprehensibility of many early computer manuals, if it hadn't been for those noble Nerds Next Door simply explaining what button to push to get the result you want, computers would never have caught on.*

If none of these roles works for you, find something that does. The idea here is to get into the right frame of mind to communicate with your audience at the right technical level.

The Writer's Duties

The fiction writer's duty is primarily to self and to art. Beyond that, depending on the writer, there may (or may not) also be a strong sense of duty to the reading audience and the paying publisher. Sometimes these duties come into conflict: write the book you want or that your readers want? Create art or a commercial success?

For both technical and UnTechnical writers, duties can conflict as well. Here are the main duties you'll be expected to fulfill, cross-purposes and all:

- **Duty to self**—your desire to do the best work you can, to write a masterpiece, to create art, and to promote your career.

- **Duty to employer**—your need to satisfy the boss for continued or future employment, and have a positive impact on the bottom line. A positive impact on the bottom line includes minimizing costs to the company by controlling the time and cost of writing, as well as the size and cost of documents and their related shipping costs. It also involves the cost of after-sales support (eliminating as many tech-support calls as possible), and improving the opportunity for future sales of updated versions and other company products. (If a customer buys a product and doesn't understand it or can't use it, they won't buy another from the same company.)

- **Duty to team**—whether you've been on the project since the beginning or just joined weeks before shipping, the design/development team has been working on the project for months or years. They deserve docs that "do right" by the product, show it in a good light, make it more useful, and promote it. (And mention all their names—properly spelled—in the credits!)

- **Duty to product**—often the product itself, whether computer game or VCR or whatever, is such a good piece of work that it deserves to be given a good chance to succeed. And that means it

should be well-documented, so it can be used and enjoyed as quickly and easily as possible.

- **Duty to customer**—marketing will identify with and represent the customers up to the point where the product is sold, but it's up to you to be there for them once they get home with it. And beyond their absolute needs, what does the customer deserve? What would you demand, expect, or want if you just brought the product home and opened the box? What would satisfy you as a customer? What would satisfy your mother (assuming your mother isn't a technological wizard)?

You may find yourself split between duties, especially between those to self, employer and customer. When this happens, try to find the common ground between duties.

Usually, you'll find that the most common common grounds are the good of the customer and the bottom line.

Nobody Reads It Anyway. Why Bother Doing a Good Job?

At least once in your writing career (probably much more often), you will either hear or make this statement.

Unfortunately, it's often true. And often deserved. Many people stopped reading manuals because they just didn't help, or weren't written for the right audience or were too boring. Or maybe a lot of people out there just can't read.

Let's face it. To many people technical documents are bad jokes. And, unfortunately, the joke's on the reader.

But it's not hopeless. During most of my time at Maxis, the management was supportive of good documentation, and my group was given a lot of leeway and enough budget to do what we thought was a great job.

Our opinion was supported by the survey mentioned in the Introduction, as well as by hundreds of letters, phone calls, emails and comments on product registration cards from readers all over the world who appreciated the work we put into our documentation. These were

people who bought a game, yet were impressed and pleased enough to call or write about the manuals.

What does this mean? It means that at the very least it's worth trying to do a good job. Who knows … somebody may actually read it, so it better be good.[5] It's like wearing clean underwear in case you get into an accident.

So what should we do? We should do the best work we can, and slowly but surely change the way the world looks at technical docs. And donate to literacy campaigns, just in case.

And if all else fails to convince you to do a good job, at least think of yourself. Your name is going on it, and if it's good, it can be a great résumé showpiece.

There is recent evidence that the world may be changing. More and more magazines that rate products, especially computers, software, and audio and video products, are rating the documentation as part of the overall package. Read the magazines in the fields you write about. See if and how they value documentation, and use it as leverage, if possible, to get the support and budget you need to do a good job. There's nothing like a review that slams your—or your competitor's— product because of the docs to get the message through to management. Also, try to find out what the reviewers liked or didn't like in the documents they reviewed.

> "Good books are the most precious of blessings to a people; bad books
> are among the worst of curses."
>
> — Edwin Percy Whipple

---

[5] *Actually, my long-term studies have shown that at least one person will read any printed material shipped with a product. And it is a universal truth that that one person will find each and every typo, error and omission and let everyone in the universe, including you, your boss, the chairman of the board of the company you did the project for, the current President of the United States, and your mother, know about them. I suspect that this person who reads everything may actually be one single person, on a mission of some sort. His name is probably Larry, although he uses many aliases. Watch out for him.*

Art and Attitude

Attitude is a very personal thing. I won't tell you how to feel about your work, but I will tell you a bit about how I feel about mine.

First, I believe that there is an art to this kind of writing, but the writing itself isn't necessarily art. If others wish to see it as art and appreciate the finer nuances of meaning, the flow of the language, the sparkling wit and all that other artsy-fartsy stuff, that's fine with me. But as far as I'm concerned, if I want to write "art," I'll write poetry, fiction, screen-plays, songs or some other set of words. When I work on an UnTechnical piece, I am concerned only with getting the necessary information into the reader's mind as quickly, easily and enjoyably as possible. (This should be sounding familiar to you by now.) Be proud of your work, but don't let that pride lead you to write more, write less or write more complexly than is in the best interest of the reader.

Art aside, the attitude that works for me as an UnTechnical writer is basically schizophrenic. I split my brain between the combination of product developer, team member, and company employee (or con-tractor) on one side and advocate for the customer and reader on the other. In this "us and them" world, the writer is both.

Know the Local Attitude About Documentation

As writers, full of the immeasurable importance of our work, we be-lieve that a product's documentation is a vital part of the product. It is actually part of the user interface, and can determine whether the cus-tomer has a good or bad experience with the product.

I've worked for and with people and companies that came close to my own way of thinking. And I've worked for and with people and com-panies with the attitude that docs are a necessary evil, and they should be slapped on as quickly and cheaply as possible. Most companies fall somewhere in between.

Contractor or employee, it is important for you to know where you and your work stand in the local psyche. It will help you to gauge your work, plan your time, estimate hours and pages, and choose your battles wisely (should you decide to choose any).

Understanding how a company pigeonholes writing also tells you about your place in the hierarchy, and your clout. Changing a company's attitude about the importance of good documentation can be done, but it is a slow process. If you wish to stay employed or be rehired on a future contract, don't push too hard.

Why push at all? Some people don't. Some people have to. It's a personal attitude and a personal choice that revolves around the question, "Are you on a mission or is it just a job?"

Be Realistic

So, OK, I've been ranting and raving and preaching about who and what you should be and do and how well you should do it. I've talked about missions and duty.

Do I always meet these standards? Heck no.

Do I try? Whenever possible.

Have I ever failed? You bet.

Am I full of crap? Your call.

Sometimes a job is just a job. Probably more often than not. So, to paraphrase Joseph Campbell, follow your bliss, but don't forget to pay the rent.

Skills and Thrills

UnTechnical Writer Skill Set

Here's a list of skills and knowledge that I used to help me hire new writers and editors for my writing department. I didn't expect anyone (least of all me) to have all these skills, but when I hired new people, I chose people who would fill out our weak spots.

Writing and Editing Knowledge/Ability

- Technical writing experience and ability
- Marketing/advertising writing experience and ability
- Creative/Story writing experience and ability
- Instructional writing experience and ability
- Editing skills and experience

Technical Knowledge

- Familiarity with:
     Windows 3.X, 95/98, NT
     Macintosh
     Novell/other networks
- Programming concepts/basics
- Art concepts/basics
- Design and layout techniques and styles

Educational Knowledge/Experience

- Teaching experience
- Knowledge of Educational Frameworks[6]

---

[6] *Many states publish educational frameworks that spell out what students should learn about different subjects at various grade levels. If you are working as a writer on products that will be used for educational purposes, whether manuals or teacher's guides, you'll want to be familiar with and refer to the local frameworks in your writing. At a minimum, you can get by with the California and Texas frameworks. Many other states use these as models.*

Software Tool Knowledge

- Microsoft Word[7] (Mac/Windows)
- Other word processors (Mac/Windows)
- FrameMaker
- Online help compilers and help systems (Mac/Windows)
- Presentation software (PowerPoint)
- PageMaker
- Quark XPress
- Pixel-based paint programs (Mac/Windows)
- Vector-based draw programs (Mac/Windows)
- High-end graphics processing programs like Adobe Photoshop (Mac/Windows)
- Spreadsheets
- Project Management software

Personal Attributes/Skills

- Sense of humor
- Self-motivation
- Good people skills
- Good management skills
- Good time-management skills
- Interviewing skills
- The ability to quickly learn new programs and procedures
- The ability to finish projects

Design Skills

- Knowledge of and experience with game or software design
- Knowledge of and experience with interface design

---

[7] *I mentioned this word processor by name not because I wholeheartedly advocate it, but because it was the company standard, and file compatibility required that all writers be familiar with it.*

Not all writing departments are going to care about all of the things on this list, but it won't hurt you or your career to know more than you absolutely need to.

Improving Your Skills

Beyond the obvious ways of improving your skills by learning new programs, reading specific books, working with a good editor and practice, practice, practice, here are a number of other activities that will help you grow as a technology writer.

"All the knowledge in the world is found within you."
— Anthony J. D'Angelo, The College Blue Book

"Never stop learning; knowledge doubles every fourteen months."
— Anthony J. D'Angelo, The College Blue Book

"No wonder I keep gaining weight."
— Michael Bremer

Read and Analyze Technical Writing

In spite of its reputation, there is a lot of good technical writing out there. Read it and learn from it. Read any technical manual or booklet you can get your hands on. Work through the tutorials. Pick a button or feature and see how long it takes you to find the explanation. Make a judgement call on the piece: Is it good? Why or why not?

As you read, ask yourself these questions:

- How is it organized?
- Who are they writing for?
- Who should they be writing for?
- What percentage is tutorial?
- What percentage is reference?
- Does the document serve its purpose? How completely?
- Can you quickly find the information you need?
- Is everything clear?

- Is anything confusing?

- Is everything consistent?

- Is anything missing?

- Is there too much fluff?

- Are the graphics useful?

- Are there enough graphics? Too many?

- How does it look? Is it easy on the eyes?

- What collateral material is there? (Quick-Start guides, readme files, reference cards, etc.)

- What about the finer details? Are there typos? Sloppy language?

- Would you be proud to have written it?

- How would you improve it?

Gather a library of good and bad samples. If you don't have the shelf space or cash to collect them, then borrow them and keep a library of reports in a binder. In each report, put answers to the questions above plus a copy of any part of the book that stands out as truly excellent.[8]

Use this library or set of reports as a resource for style, tone and organization whenever you start a new project.

Read Good Popular Science Books

These days, popular science books are doing well. There are a lot to choose from. And many of them contain wonderful, interesting, clear explanations of complex topics.

Read at least a few of these books, especially those concerning subjects you know nothing (or very little) about. The less you know going in, the better you'll be able to judge how much you learned, and how well new concepts and systems were explained.

The differences between popular science books and product documentation include:

- the audience (generally popular science readers are more interested in science and technology than the average consumer),

- the writer's ultimate goal (provide a good read and entertain rather

---

[8] *Of course, by "copy," I mean copy by hand. I would never, ever suggest photocopying copywritten material.*

than get the information from point A to point B as quickly as possible), and

- the structure (closer to a novel than a reference).

Differences aside, there's much to be learned from this writing. Notice the enthusiasm for the subject that many of these books project. See if you can get a little of that enthusiasm into your writing for the product you're working on (but don't overdo it).

Screenplays and Format

Something to know about screenplays is that they have a very specific format. Type size and spacing and margins are all planned out so that one page of screenplay equals approximately one minute of screen time. Think about how much can happen on the screen during a minute of a movie. With a few exceptions, that fits onto a single page. You have to know what to leave out, what to leave in, and how to make it read well with very few words.

Here's your chance to be petty! You can choose whether to be jealous of screenwriters who don't have to worry about designing a format for every piece of work they do, or be smug and superior because their hands are tied and yours aren't.

Read Screenplays

Have you ever read a good screenplay? Try it sometime.

What's amazing about a good screenplay is how complete it is in spite of what isn't there. Unlike a novel, there's no 10-page passage describing how, as the breeze moves the curtain, the sunbeam coming through the window changes shapes. There's no deep description of the location or clothing or even characters unless they play an important role in the plot. There's nothing but the bones of the story: just the action and dialog and just enough more that makes it a good read.

It's almost skimpy. Just enough to tell the story and give you the feel, the mood. But it's compelling.

No, UnTechnical writing shouldn't read like a screenplay. But there's a lesson in craft to be learned there: cull down your prose to what's important, to only what's necessary. Present steps and actions in a specific sequence, making sure the proper information is there when the reader needs it. And, in spite of the sparseness, keep it entertaining through active language, and subtle humor.

Once hard to find, other than from college libraries, film schools, or just about anyone living in Los Angeles, screenplays are now available in many bookstores and libraries. And you can also get a lot of scripts off the web.

Read some screenplays for movies that you like. Take a screenwriting class if you can, or read one of the seven billion books about it. You'll learn a lot. And you'll have fun rounding out your writing skills in new areas.

Read Good Children's Books

Many great examples of clear, concise, and simple-yet-compelling writing can be found in children's books. In particular, those written for the Juvenile audience (book-publisher talk for 8- to 12-year-olds) are a good resource.

No, your book shouldn't necessarily be written for 8- to 12-year-olds (unless that's your target audience). But reading these books, and seeing that it's possible to tell stories with all their drama, passion and action—in a shorter format, using a simpler vocabulary—can have a positive effect on your writing.

Go back and reread the books you loved as a child. Talk to kids and find out which books they like, and why. Most librarians and bookstore employees can help you pick out some good samples.

> "I wrote a few children's books ... not on purpose."
>
> — Steven Wright

> "You know how it is in the kid's book world; it's just bunny eat bunny."
>
> — Anonymous

Work on Your Interviewing Skills

Interviewing your team members and other local experts on the project is a valuable way of gathering information for documentation.

Interviewing goes beyond mere conversation in that you prepare yourself with questions and a list of subjects to discuss. You listen carefully,

and make sure you understand everything, and keep asking questions until you do. If a new subject that you want to cover is brought up, follow through with it, then return to your original agenda.

Unless you're a born multitasker with great shorthand skills, a small tape recorder can make interviewing much better and much faster. Always test your recorder to make sure it works and that everyone's voice is coming through clearly. If the recorder uses batteries, change them regularly. Running out of a meeting to get new batteries, then asking your subjects to start over can be embarrassing.

Even if you have a tape recorder, bring paper and pencil. You or the interviewee may need to make drawings or sketches to illustrate points, and those don't come through well on tape.

Learn Interface Design

"Damn it, Jim, I'm a writer, not an interface designer!"[9] That's what the good space-faring TV doctor would say in your place. Yet we all know that before the next commercial, he'll have that interface designed and the day saved.

Yes, writing and interface design are two separate tasks, usually with separate job titles, but they're closer than most people think.

The interface—and this goes for hardware and software—is the part of the product that the customer sees, feels and uses to make the product do what they want it to do. It separates and protects them from the complex inner workings, while enabling control over those workings.

Most of UnTechnical writing is explaining how to use the interface to control the product. A manual for a word processor doesn't need to teach how the computer stores the characters in memory or on disk. It does need to teach how to interact with the interface (onscreen buttons, menus and dialog boxes) to make the internal workings of the word processor do what the writer wants. A manual for a television doesn't explain how a television works, it tells the customer how to interact with the interface (the remote) to make the TV display the desired show at the proper volume, brightness, etc.

---

[9] There may be a law in some states that requires all technically-oriented material with any sign of a sense of humor to have at least one Star Trek reference.

As a technical or UnTechnical writer, you will study, analyze, dissect, and describe in minute detail many, many user interfaces. You will learn about good and bad interface design—especially bad—even if you don't want to. If you actively seek information about interface-design, you'll get a lot more exposure to the good side of things, and, perhaps, improve your career. If it's known that you have interface design skills, you'll be brought into projects earlier, which not only means more contracting hours, but also gives you a chance to make a positive impact on the product.

Warning! Prepare for a minor rant. It is my hypothesis that fully half of the reason technical writing has such a bad reputation is because the interfaces we write about are bad. Garbage in, garbage out. No matter how good a writer you are, how dedicated and caring you are, how diligently you write, edit and test your work, if the product's interface sucks, your manual will suck. OK, rant's over.

Interface design is a big subject, and the chapter on it later in this book will give only a basic introduction to the parts of interface design that writers really should know.