



Introduction (a.k.a. “Documentation: Why?”)

“An hour spent documenting at 2 p.m. may prevent an urgent call at 2 a.m.”

Documentation is ubiquitous in our field. It seems to expand until it fills every box, binder, shelf, and flat surface. We’re awash in READMEs, HOWTOs, and FAQs. Thanks to the Internet, we’ve got access to bug-tracking databases, discussion groups, and the Web sites of most vendors and standards bodies. It would seem we have a handle on the whole documentation thing.

Or do we?

Having so much information so readily available lulls us into a false sense of security. We know that within a few moments we can locate a document—either digital or dead-tree—that will tell us all we need to know about a given topic or procedure. We’re confident in our ability to apply this information to the task at hand. We’re comfortable in the knowledge that, with enough research and some quick thinking, we can solve just about any problem.

There are a few flaws in this train of thought.

- You need to document your specific environment and needs. Vendor-supplied documentation tends to address generic situations, rather than specific implementations. Even when vendor-supplied information is specific, it rarely describes your problem perfectly.
- Discussion groups, bug databases, and similar resources offer incomplete, contradictory, and often incorrect information. In many cases, not enough context is supplied to determine the applicability of the information to your situation.
- Documentation most often explains how to do things right, rather than what to do when things go wrong.
- The most critical information remains undocumented and may reside with one or a small number of individuals, who become single points of failure.
- The documentation you find may not mesh well with your own processes and procedures. Identical tasks may be completed in different manners by different individuals. You need documentation that directly fits your particular needs, rather than having to translate from another person’s way of doing things to your own each time the document is used.
- External references, particularly those that aren’t printed, may be unavailable when you need them.

2 / Introduction

- The information may be available, but in a form that's ineffective, poorly organized, incomplete, or inappropriate for your needs.

My goal is to provide you with solutions for these and similar problems. By the time you reach the end of this booklet, you should understand the need to incorporate the practice of documentation as part of your daily routine, and you should be armed with the knowledge necessary to produce effective documentation that will become an integral part of your organization's approach to IT.

But Nobody Reads Documentation!

You see documentation everywhere. You've probably got (literally) tons of it yourself. Chances are good that you're the type who installs first and reads the fine manual (RTFM) later. Our professional culture is rife with jokes about the misuse and disuse of documentation. Why is that?

Perhaps because much of the documentation we must deal with on a daily basis is nonexistent, poorly written, poorly organized, and/or poorly presented. In other words, it's not useful, accessible, accurate, and/or available (our friends, the Four Keys to Good Documentation).

I've heard sysadmins—particularly junior sysadmins—complain that manpages are neither useful nor accessible. I've watched them spend many minutes puzzling over a manpage looking for the answer to a question, only to come away from the experience empty-handed and somewhat bitter. Does this mean manpages are poor documentation? Hardly! But it means that they were the wrong tool for the job, from two perspectives. First, the person may not have known where to look for a better solution or may not have had other resources available. Second, the manpages were written for a particular audience, which did not include the junior admin.

I've seen users shun ticketing and problem-tracking tools in favor of a quick email message to a sysadmin, a phone call, or the dreaded "line of users at your door" approach. Is it because ticketing systems are poorly designed, inaccurate, or otherwise useless? Not a bit. Many times, it's because the interface is confusing and/or time-consuming. People tend toward the most direct solution to a problem. When your documentation seems more of a hindrance than a help, people will go out of their way to avoid using it.

You can escape these and other problems by following five simple rules when setting out to write your documentation:

1. Know your audience.
2. Know your content.
3. Know your requirements.
4. Make it a habit.
5. Advertise.

1. Know your audience. I can't stress this point too strongly. Know who will be reading your documentation. Know what they expect from it and how they intend to use it. Know what they need to get out of it.

2. Know your content. Know what type of information you're going to present and what level of detail is appropriate for your audience. Proficiency in the subject matter is extremely useful. If you find you know little more about the subject than your intended audience does, it may be in everyone's best interest for you to delegate the work to someone with a firmer grasp of the material: the better the writer understands the subject, the better the chances of avoiding jargon and overly complex explanations.

Of course, the opposite can hold true as well—subject-matter experts are sometimes prone to forget their audience, going into complex, jargon-heavy explanations that quickly become far too esoteric for the audience, which goes to show why knowing your audience is such a crucial factor in writing good documentation. If you know your audience and identify a subject-matter expert who's good at describing things to the people in that target audience, your documentation project is likely to prosper.

3. Know your requirements. Before you sit down to start writing, or design your delivery format, or (worst-case scenario) start telling everyone in your organization about your fabulous new documentation resource, you should define your requirements clearly. What do you need to put into the effort? What does your audience need to get out of it? How do they need to get it? What are your constraints on delivering it? Once you've answered these questions, you'll have a clearer picture of what it is you need to document, how it should be documented, and how it should be delivered.

4. Make it a habit. All documentation should be viewed as “living” documentation. Once it's written, it shouldn't be tossed aside and forgotten. I can promise you that sometime, somewhere, someone will find it and try to use your documentation. If you haven't been maintaining it, it will be worse than useless—it may be dangerous (ever rewire a three-phase circuit and forget to tell anyone?). Any failure to maintain your documentation will be quickly discovered by your audience, who will henceforth ignore your efforts.

For example, let's say you've got an online collection of frequently asked questions (FAQs) for your group. You decide to make this collection available to the rest of the organization, as a way of minimizing your support calls. That's certainly an admirable goal! But let's say 5–10% of the questions and answers are out of date, incorrect, and/or poorly worded. Your users won't ignore those bits of information; they'll interpret that sort of inaccuracy as reflecting badly on the entire documentation system. A system like the one I just described would quickly fall by the wayside, rarely to be used by its target audience. Your audience expects and deserves accuracy and completeness, and they'll quickly turn on you if they don't get it. Documentation is perceived as authoritative. To insure that it remains so, you must work to make the maintenance of your documentation a regular habit. If you've documented a certain system, or component, or area of knowledge in your organization, you should always remember to update the documentation as soon as a change is made to whatever's been documented. In fact, you shouldn't consider your work complete until you've not only finished the technical part of the job, but finished documenting the work as well.

5. **Advertise.** So you've been hard at work, analyzing your needs and those of your target audience, and you've constructed a miracle of modern documentation, the very model of successful sysadmin documentation effort. Why are people still bugging you with questions about things that are clearly explained in your documentation? It could be as simple as this: they don't know the material exists. Once you've finished your documentation and made it available, you must tell people about it, particularly if it's documentation that's not taking up a significant portion of their desk (the satisfying thud when it arrives is usually a dead giveaway for that type of documentation). If people don't know information is available, or how to access and use it, they can't take advantage of it. So tell them! Let them know that this new resource exists. When you're asked a question that is covered in the documentation, refer them to it rather than answering the question yourself. Help your audience help themselves. Excellent documentation nobody knows about is just as useless as poorly written, poorly presented, or nonexistent documentation.

Preview

The chapters that follow will introduce you to several types of information you may wish to document (documentation content), describe various tools and techniques for documentation (documentation format), provide you with a procedural framework to get started with documentation in your organization (documentation methodology), and offer a series of examples meant to illustrate the utility of certain types of documentation.

Throughout the booklet, the terms *system administrator* and *sysadmin* are used interchangeably. The abbreviation IT, standing for "Information Technology," is meant to encompass all areas of a sysadmin's responsibilities.

Presuppositions

I assume you're an active system administrator with a need for or interest in documentation as part of your job, and/or are responsible for documentation of the system administrator role in your organization. I also assume that you'll use this booklet to guide your documentation efforts.