

## Defending The Documentation Budget

*Douglas C. Shaker*

Money is tight these days, especially in the software business. When money gets tight, budgets get cut. Unfortunately, sometimes the wrong line items get cut. Far too often, software companies cut the documentation budget when they shouldn't. This is because most software companies think of documentation in the wrong way.

Nearly everyone—VPs, engineers, sales reps—thinks of documentation as this thing that you need to have so you can say that the software is done. Some even recognize that it has this niggly, bean-counter purpose of reducing support. However, both of these views miss a critically important function of documentation. One of the most important uses for documentation is selling the software—not in some vague, this-might-be-OK, eye-candy flyer way, but in a more concrete, get-the-order way.

Think about the typical sales process. Your company has given someone a great demo and it has gone well, but everyone (you and your prospect) knows that demos can be a perilous walk down a thin path of working software, with a swamp of bugs and unimplemented features on all sides. Your potential customer is interested in committing some bucks, maybe even making a big buy, but first they want some time alone with the software to they can push on it and see what works and what doesn't. And, they don't want you there. So you have to leave them with the software and the docs.

The customer gives the software and the docs to a staff member to evaluate—not their best person, because they're too busy, and maybe not their worst, because they're not sure if they'll get an accurate evaluation. So the customer assigns the evaluation to an intern as a test to see how robust the software is. They give the intern instructions something like this:

Take a look at this software in the next day or two and see if you think it will work for us. Don't take more than 4 or 5 hours. I need your opinion in an email by Wednesday at 1.

Now the intern sits down with the software and thinks, "How am I going to get this done with all the other crap that needs to be done by Wednesday?" And, if he wants to he can do NOTHING with the software and just say to the manager, "Naw, I don't think so. There are too many bugs and it looks like it was designed ten years ago," and the boss will go with that. This is where good documentation can come into play to make it easy for the intern to give the software a proper evaluation.

The first thing the docs need to do is get the intern to install the software. When he's trying to decide if he can get the install done before lunch, the install guide needs to tell him "Sure! It'll just take five minutes!" so the install document needs to be easy to find, clearly labeled, and it needs to tell how long it is going to take to do the install.

When the software is installed, the docs need to make it easy for the intern to come up with good info to give his boss. He should be able to pick up the docs and find the software features he's looking for in five minutes. Then, the docs should show him that the software designers understand his problems and how the software can solve them. The docs need to present all of this information in a clear and easy to follow manner.

Within the first fifteen minutes, he should be thinking, "This looks pretty good." Within the first hour, he should have an idea how to answer the boss's questions. And within two hours, he should be able to try a solution and see that it works. At the end of the four hours, he should have tried a couple more test runs, found out

that the manual isn't lying, and then be looking in the manual for other features that will make it a better fit for their needs.

If the software and docs pass this test, chances are the customer will want to buy the product. After they become a customer, they won't switch to something else unless you can't provide new features they need. But the docs will have fulfilled one of their most important functions—they will have gotten you a sale.

If you think your documentation budget is being cut in a way that can limit your product or make it harder to sell, try introducing your marketing folks to the idea of documentation as a marketing tool, one that stays with the potential customer and sells them on the features of the product.

And, if you need help getting the docs done, give Expert Support a call. We understand software, we understand how people really use software, and we understand documentation. We can create superior product documentation for your software that can help your software succeed.

---

Douglas C. Shaker is Expert Support's Vice President.

---

This article is part of a series of articles on technical writing that Expert Support hosts on its website, [expertsupport.com](http://expertsupport.com). Expert Support is located in Mountain View, California and supplies contract technical writers to the computer and software industry.

