

# Fantasies of the Technical Writer

By Andrew Swartz, (first published 17 May 2004, UsabilityNews.com)

*"I'm afraid I'm going to give the pub a miss tonight, there's a new manual I want to read."*

*"Early night tonight for me. There's a help system I can't wait to dive into."*

These are words, alas, that the average technical writer will never hear. It may be grossly unfair, but the sad fact is that most of us will do just about anything to avoid opening a manual or thumbing through a help system.

I spent years writing manuals and help systems, and have the utmost respect for the profession. On the whole, it is a startlingly dedicated group, working hard, and often for little reward or respect. And saddest of all, for many writers, there is a constant nagging feeling that the fruit of all the labour is going to waste. Is anyone reading it?

Of course, some people do read it. Programmers and engineers are brilliant technical readers, but most of us leave the manuals suffocating in their shrink wrap. And then there's The Martin Effect. You know, there's the one friendly guy who actually knows how to do mail merge and make tables, and doesn't mind explaining it. In one of my offices, that was a fellow named Martin, and he was the one who read the manuals. If those manuals were good, we all benefited.

So well done, all you technical communicators. Your work may not be as popular as Tom Clancy, but it does make a difference, and the people who matter do notice. But I'm not going to let you off the hook completely. You need to do more, and get more involved with your products.

To explain, here are some findings from some studies I've been involved with:

- Some years ago, I wrote manuals for a specialised software product. To study how effective the manuals were, I visited the offices of the users, hoping to find information such as whether the table of contents was used more, or the index, whether the callout style in screen shots was effective, and if the tasks were clearly laid out. What I found instead was that no one could find the manuals. Either they were locked in a cabinet for safekeeping and no one had the key, or they were lined up on a shelf with lots of other manuals, and every one of them had the same black wire binding. It took 10 minutes just to find the right manual.
- At Apple, we studied a help system in a user lab by giving people tasks so difficult they couldn't possibly figure them out without resorting to looking up the answers in help. In the first days of two hour sessions, users never once turned to help, despite our ever more desperate pleas and hints. People, it seemed, would rather suffer hours of frustration than open a help file.

What's going on here? The usability issues in technical writing are still at a more basic level than most of us acknowledge. The problem is that the writing is too far away from the action. The instructions aren't where the users are – and we users, frankly, are much too lazy to go find the instructions when we need them.

So here's my basic argument: for the most effective user experience, reduce the distance between the instruction and the action.

To understand, imagine one of those doorknobs that usability guru Don Norman writes about. Norman points out that the design and look of a doorknob communicates its function. Some handles just want to be pushed, and others demand to be pulled. And, on an otherwise slow day, you can have hours of amusement watching a door with the wrong kind of handle make person after person stub their toes or dislocate their shoulders.

The best thing to do with a door that has a bad handle is to replace the handle. (This is a metaphor. As I write 'handle' or 'door knob' you should be thinking of 'software' and 'dialogue boxes'.) But sometimes this is not possible. To help the poor door user, you need some kind of written communication – some words or pictures to tell people how the door works.

Now if we were to take the typical technical writer approach, we would do this by shipping a short manual with the door, and hope that everyone approaching the door would flip through the manual to find the instructions about pushing and pulling, and there would be about as much chance of that happening as your neighbour reading the manual for his new computer.

The best place to put the instruction is on the handle itself – a short and eloquent word would do – PUSH, or PULL.

(If you want a good live illustration of this, ride one of the new British trains with wheelchair accessible restrooms. Their doors open and close using pushbuttons, and the buttons are usually on the wall opposite the door. When these trains first came out, you could watch one person after another struggle to close the door by hand, and many would give up because the instructions were so far away from the door.)

The same rules apply to software instructions. If you are writing instructions for a consumer-oriented product, especially discretionary interfaces on the web, do everything you can to bring your instructional words and pictures closer to the audience. There are four types of distance, both virtual and real, to consider reducing:

- **PRODUCT DISTANCE.** The best option is to improve the product so that it is obvious how it works – in the door example, it's to make sure that the handle tells the user correctly whether it is a PUSH or a PULL door. While technical writers typically don't see product design as their role, often they are the first people to see how the product will look to naïve users and so are well placed to lobby for product changes.
- **GEOGRAPHIC DISTANCE.** Make sure your instructions are physically close to the interface they describe. Put prompts next to the text boxes they describe, or at least on the same page. Every pixel matters.
- **CLICK DISTANCE.** If you can't get the text on the same page as the interface, reduce the number of clicks between where the user notices the problem and where they find the solution.
- **CONTEXT DISTANCE.** Users have a great deal of problems learning special help interfaces. Make sure the rules in the help system are either already familiar to the user, or are similar to the rest of your product.

In short, stay close to the user.

I once worked on a new inkjet printer, and discovered that my company was getting thousands of support calls that their new printers weren't working. In almost every case the cause was the same – users had failed to remove the sticky orange tape from the printer cartridge before installing it. The manual of course described this in detail, but these people didn't read the manual. We moved

the instructions to a small bit of card stock which was incorporated into the tape that needed to be removed, and the problem went away. Thousands of customers happy, thousands of dollars saved in customer support, and our reputation for quality intact.

That's what can happen when a technical writer moves beyond the bindings of a book.

### ***About ExperienceLab***

ExperienceLab (formerly Serco Usability Services), are a global experience design research agency. They help organisations optimise their customer experiences, from web to TV and mobile, from advertising to physical environments. They've been doing this for a while, pretty much since the first computers and networks were created, so they know a thing or two about how to make people, processes and technologies work in harmony.

ExperienceLab use a wide range of techniques to tailor a research solution that fits your business objective, including ideation sessions, proposition analysis, customer needs mapping, usability testing, benchmarking and touch point integration studies. As a co-founder of the UXalliance we also provide research on a global scale.

Why not visit the ExperienceLab blog ([www.experiencelab.info](http://www.experiencelab.info)), which features the latest thinking on experience design issues.

#### **Serco ExperienceLab**

22 Hand Court  
London  
WC1V 6JF

+44 (0)20 7421 6499

[info@experience-lab.co.uk](mailto:info@experience-lab.co.uk)

[www.serco.com/experiencelab](http://www.serco.com/experiencelab)